
Základy digitálního zpracování obrazu

RNDr. Josef Pelikán

Kurz vznikl v rámci projektu "Rozvoj systému vzdělávacích příležitostí pro nadané žáky a studenty v přírodních vědách a matematice s využitím online prostředí", Operační program Praha – Adaptabilita, registrační číslo CZ.2.17/3.1.00/31165.



Projekt A-NET je financován Evropským sociálním fondem, rozpočtem ČR a MHMP.

"Praha & EU: Investujeme do vaší budoucnosti."

Základy digitálního zpracování obrazu

© 2010 Josef Pelikán, CGG MFF UK Praha

<http://cgg.mff.cuni.cz/~pepca/>

Ke Karlovu 3, M3, 3. 11. 2010, 9:00 - 12:10

Plán přednášky

- ♦ **základy vnímání barev**
- ♦ rastrový obraz a jeho pořizování
- ♦ barevné systémy, barevné operace
- ♦ průhlednost a kompozice obrazu
- ♦ histogramy a jasové operace
- ♦ konvoluce a lineární filtry, rank-filtry
- ♦ další efekty na rastrových obrázcích
- ♦ HDR grafika, pořizování a prohlížení HDRI

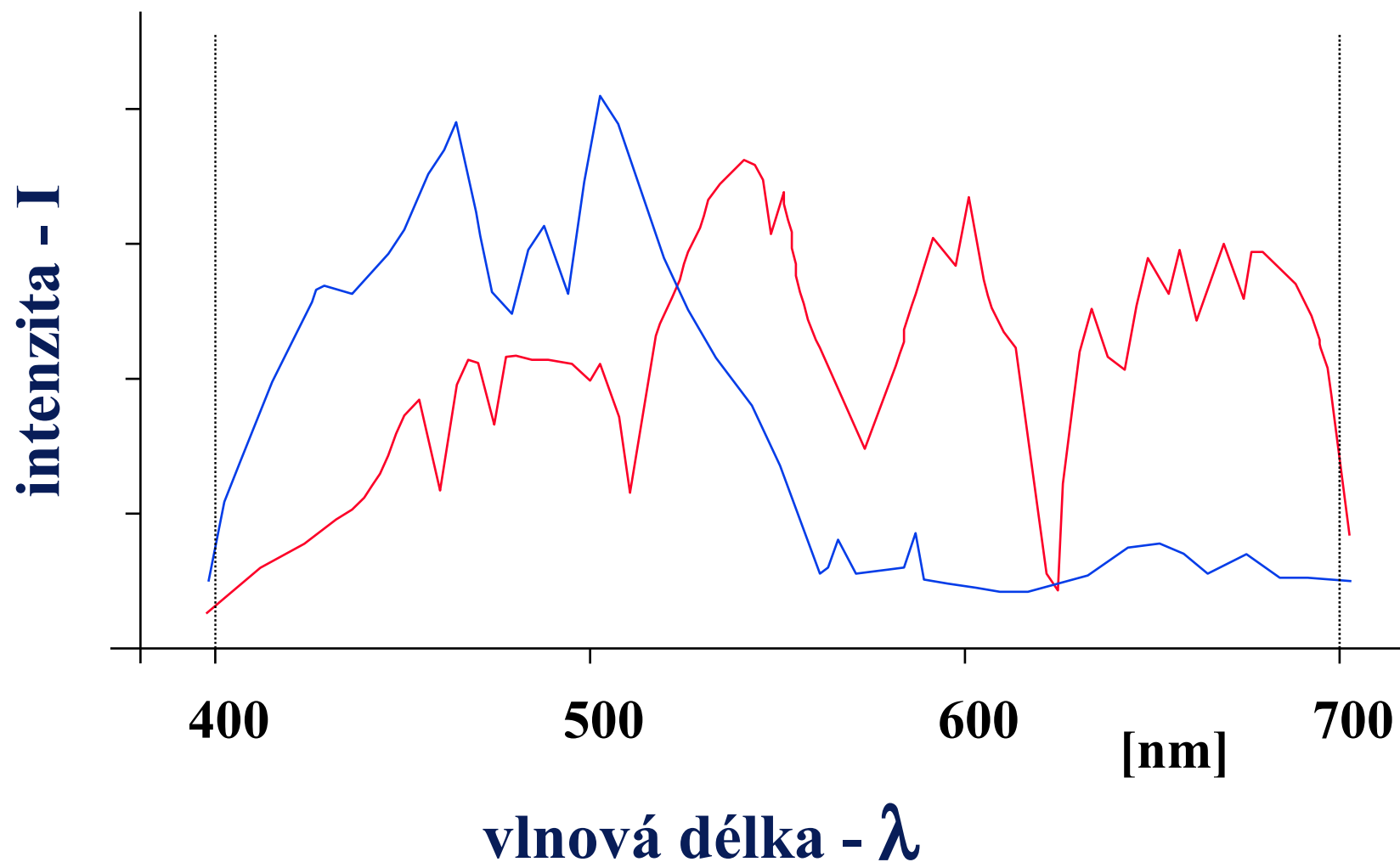
Co je to světlo?

Špatnota bludy tvořitelská:

Newton stvořil blud, že Sluno vysílá ze sebe jemné částčky proti Huyghensovým ukám, že světlo jsou chvěje tenýra zrakovým čivem pojaté ...

(Jakub Hron: “*Skutky lidské, čili Jeden tisíc špatnot žijby a konby lidské*”, 1907)

Viditelné světlo, spektrum



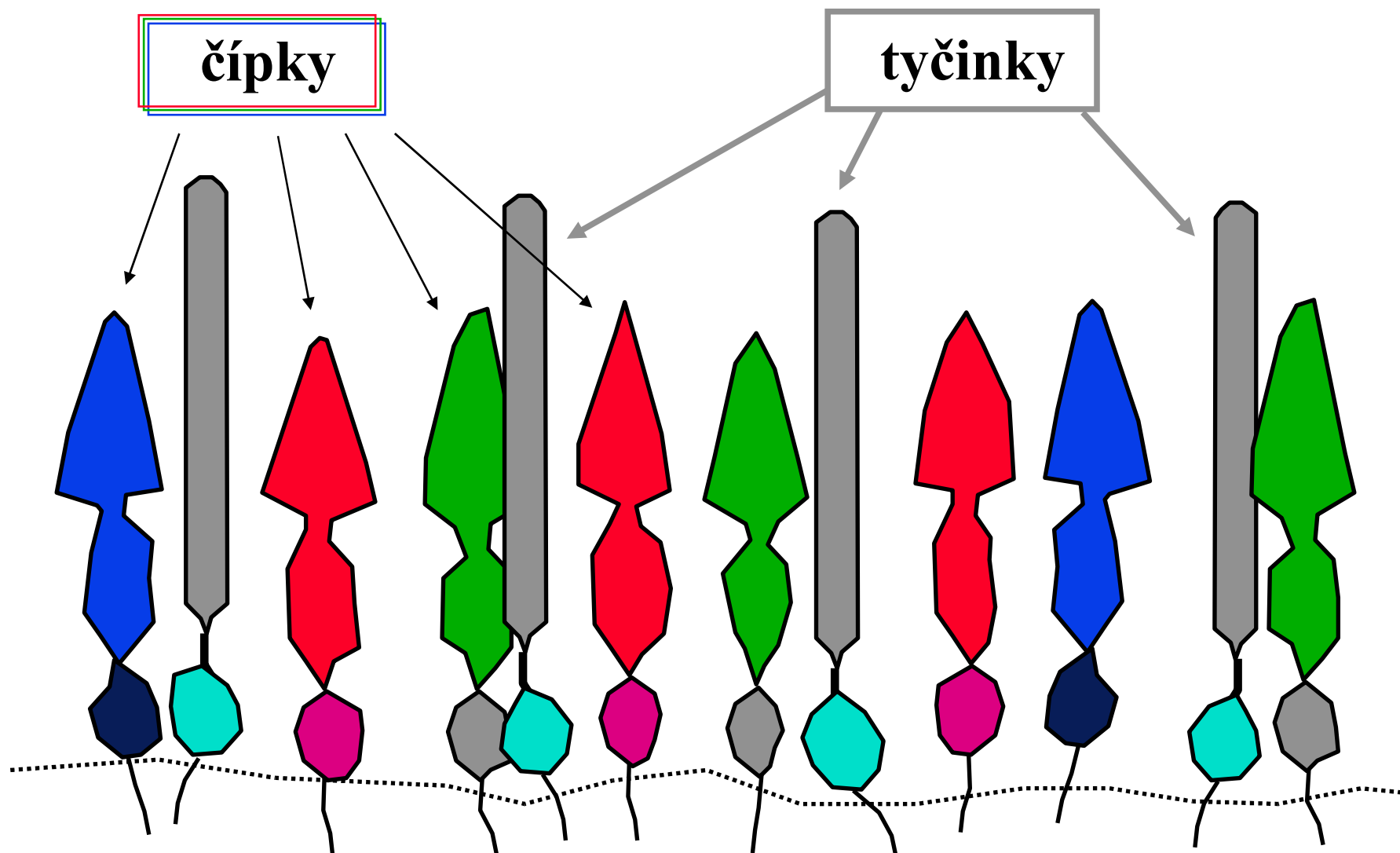
Barevný vjem

- ◆ prostor všech spekter má **nekonečnou dimenzi**
 - systém lidského vidění je však nedokáže všechny rozeznat („metamery”)

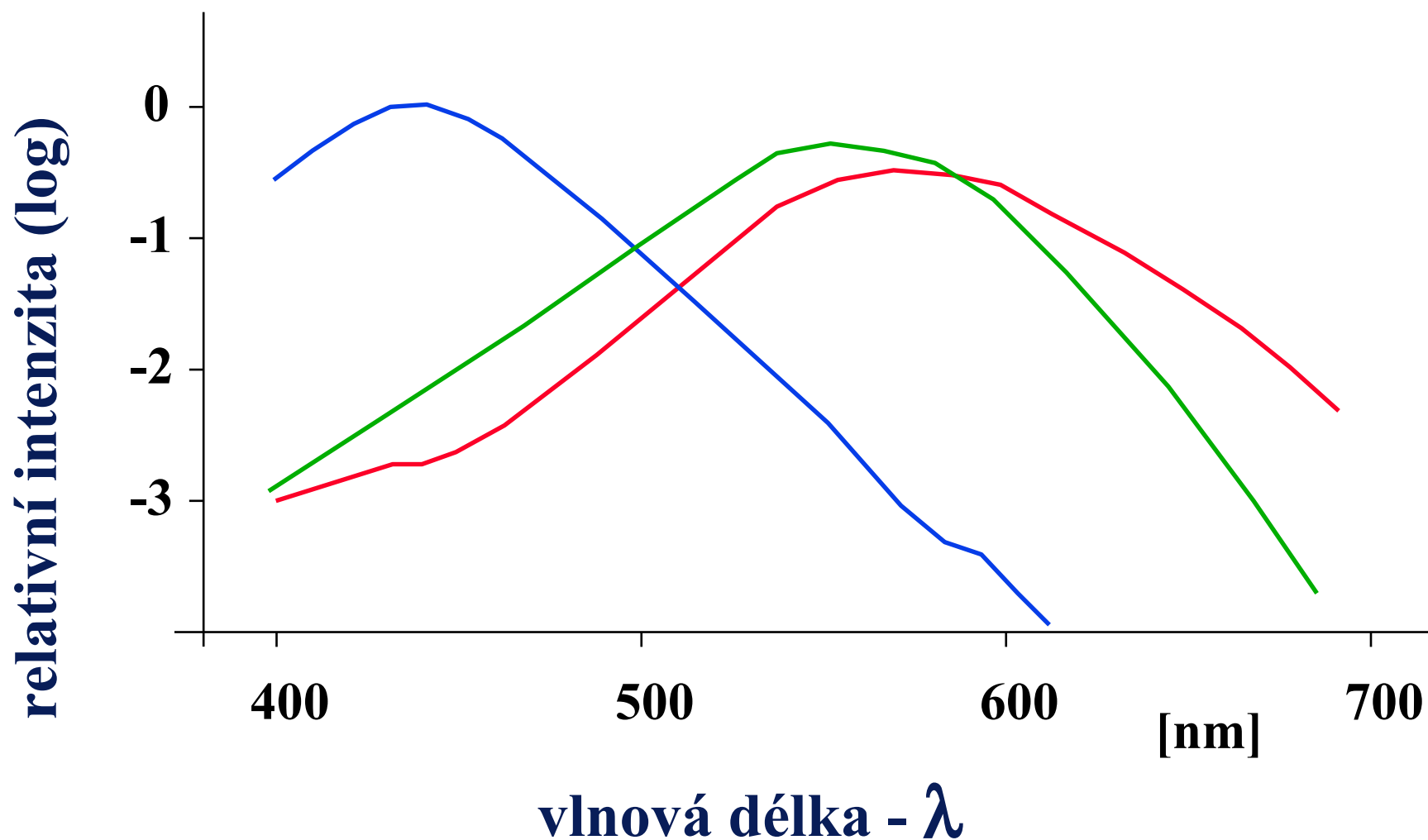
- ◆ **Grassmanovy zákony (1854)** - lidské oko vnímá:
 - **dominantní vlnovou délku** (odstín, „hue”)
 - **čistotu barvy** (sytost, „saturation”)
 - **intenzitu** (jas, „brightness”)

barvy lze aditivně skládat ($A = B, C = D \Rightarrow A + C = B + D$)

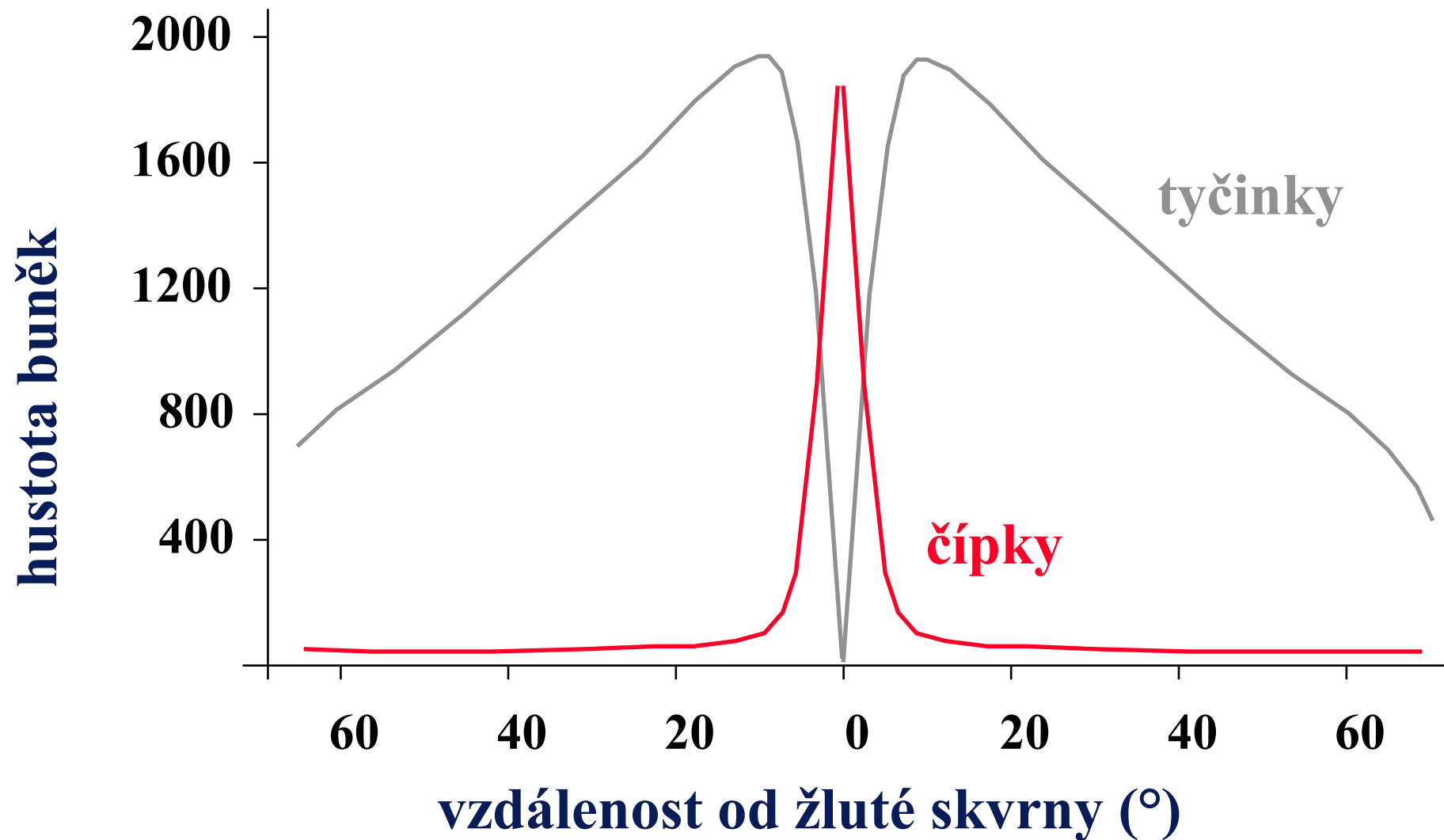
Sítnice



Tři fotopigmenty



Rozložení fotoreceptorů



Vlastnosti systému vidění

- ◆ různá citlivost na **červenou** (0.3), **zelenou** (0.6) a **modrou** (0.1) barvu
 - navíc střed žluté skvrny téměř neobsahuje „modré” čípky
- ◆ zaostřuje se **podle jasové složky** ($Y = R + G$)
 - nelze dobře zaostřit na rozdíly v modré složce
- ◆ větší citlivost ve **svislém a vodorovném směru**
 - až o 30% větší rozlišovací schopnost
 - evoluce, kulturní vývoj..

Rastrový obraz

- ◆ základy vnímání barev
- ◆ **rastrový obraz a jeho pořizování**
- ◆ barevné systémy, barevné operace
- ◆ průhlednost a kompozice obrazu
- ◆ histogramy a jasové operace
- ◆ konvoluce a lineární filtry, rank-filtry
- ◆ další efekty na rastrových obrázcích
- ◆ HDR grafika, pořizování a prohlížení HDRI

Obrazová funkce

- ◆ „okno“ do reálného spojitého světa
 - zobrazení $\mathbf{R}^2 \rightarrow$ „barva“
 - nekonečně zvětšovatelný obraz
- ◆ diskretizace obrazu
 - vzorkování roviny v pravidelné mřížce
 - matice pixelů
 - praxe: snímací sensor fotoaparátu, kamery
 - druhá diskretizace – hodnoty pixelů (viz později)

Rastrový a vektorový výstup

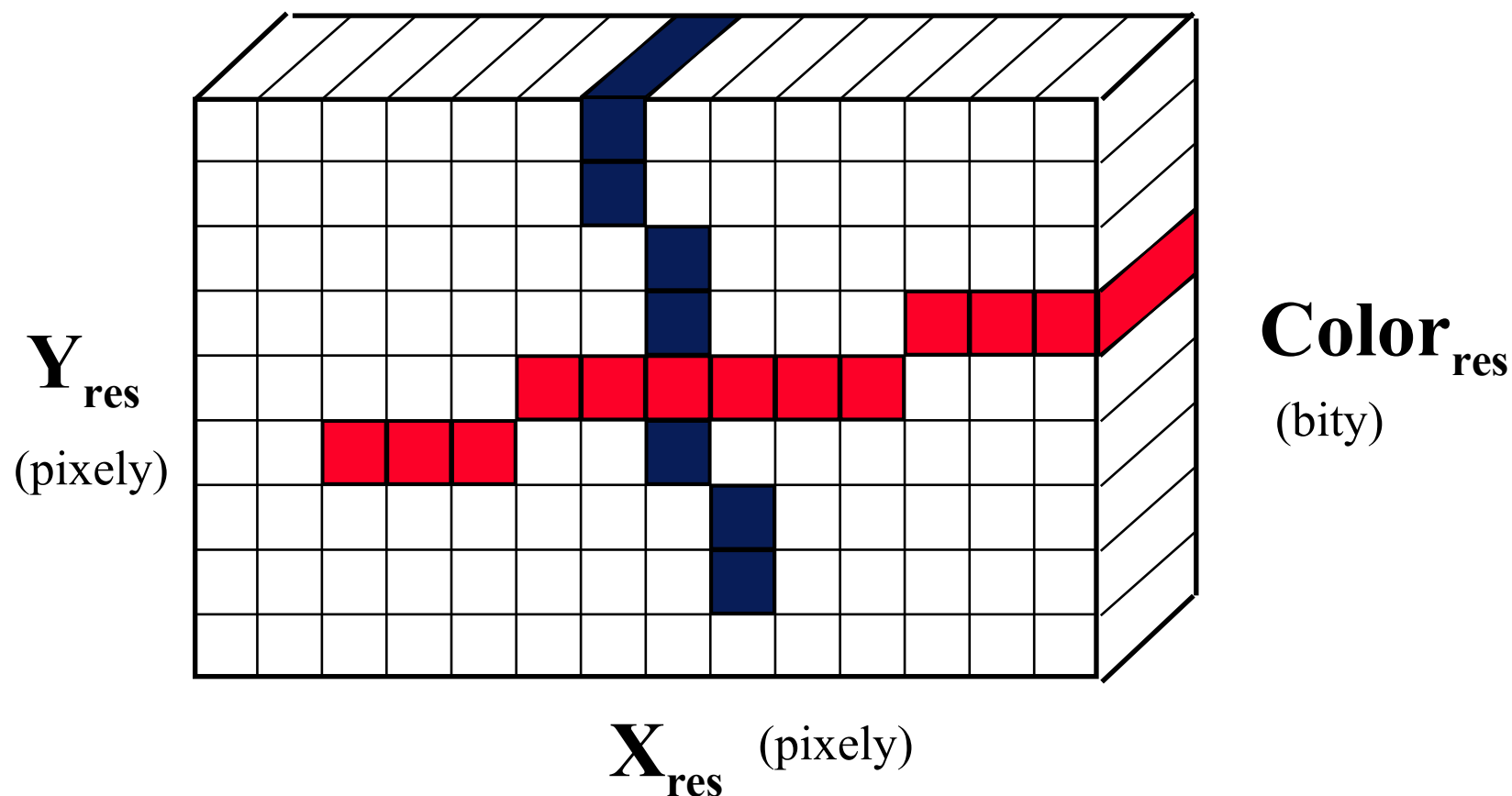
♦ rastrový výstup:

- jsou přímo ovládány (adresovány) jednotlivé pixely
- data jsou závislá na rozlišení (nelze je jednoduše škálovat)

♦ vektorový výstup:

- zobrazují se přímo složitější objekty (čáry, křivky, písmo, plošné útvary)
- data nejsou závislá na rozlišení (lze je škálovat až v zobrazovacím zařízení)

Rastrový obraz



Např: $640 \times 480 \times 8$ bitů, $1600 \times 1200 \times 24$ bitů

Formáty pixelu

❖ celočíselné hodnoty

- starší (klasický) přístup
- obvykle 8bit, $3 \times 8\text{bit}$ nebo $4 \times 8\text{bit}$

❖ plovoucí desetinná čárka

- HDR grafika („High Dynamic Range“)
- obvykle $3 \times \text{float}$ (96bit) nebo $3 \times \text{half}$ (48bit)
- bez problémů se ztrátou přesnosti..

Snímání rastrového obrazu

- ♦ **digitální fotoaparát / kamera**
 - plošný snímač (CCD, CMOS)
 - barvy → nejčastěji Bayerova maska
 - náročné vyčítání dat
 - syrová data (RAW) se přímo v přístroji zpracují
 - » specializovaný digitální obrazový procesor (DIGIC..)
- ♦ **scanner** (filmový, stolní, kopírka)
 - obvykle **lineární snímač** (1D)
 - jednodušší vyčítání, ale nutnost mechanického pohybu

Plošný snímač obrazu

✦ rozměry a rozlišení

- ✦ větší fyzické rozměry → méně šumu (fyzika)
 ... i optika bývá kvalitnější
- větší rozlišení (více MPix) → více šumu

✦ citlivost snímače (ISO)

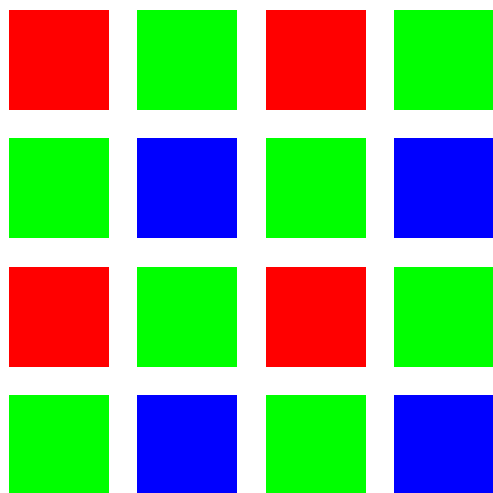
- pouze nastavuje **zesílení** při ADC konverzi
- větší citlivost (zesílení) → více šumu

✦ snímání barev

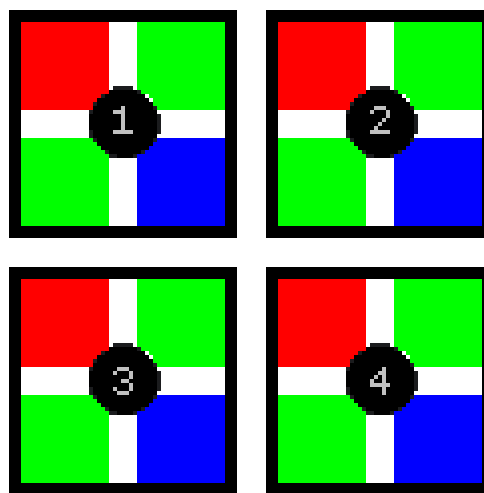
- Bayerova maska, přímé uložení do RAW

Bayerova maska

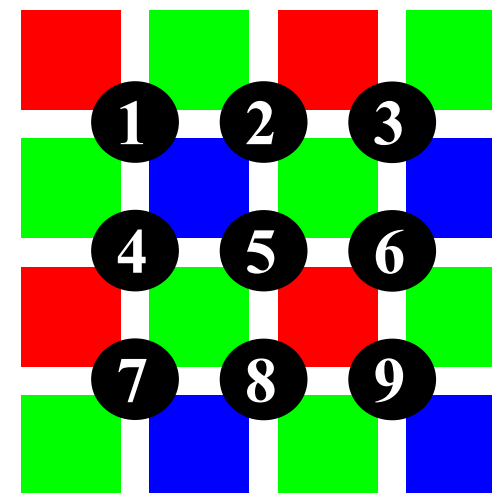
- barevné filtry pro RGB
 - jednotlivé složky se snímají odděleně (vedle sebe)
 - menší efektivita snímače, ale jednoduchá výroba



maska senzoru



přepočít na pixely

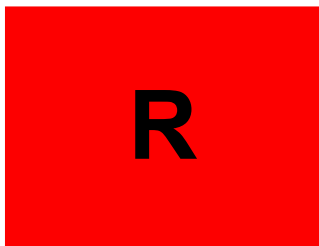


efektivnější
přepočít

Barevné systémy

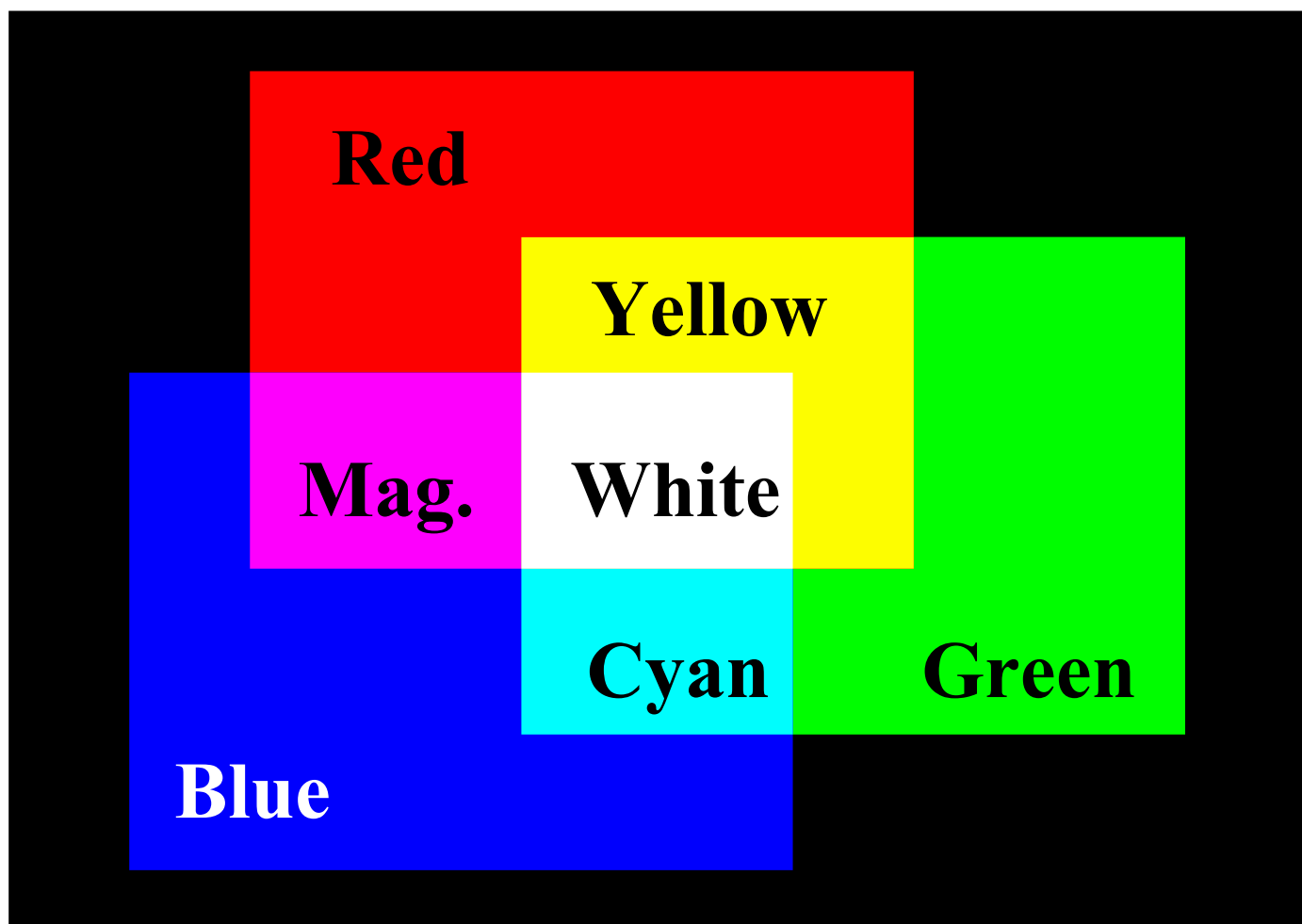
- ❖ základy vnímání barev
- ❖ rastrový obraz a jeho pořizování
- ❖ **barevné systémy, barevné operace**
- ❖ průhlednost a kompozice obrazu
- ❖ histogramy a jasové operace
- ❖ konvoluce a lineární filtry, rank-filtry
- ❖ další efekty na rastrových obrázcích
- ❖ HDR grafika, pořizování a prohlížení HDRI

Barevný systém RGB

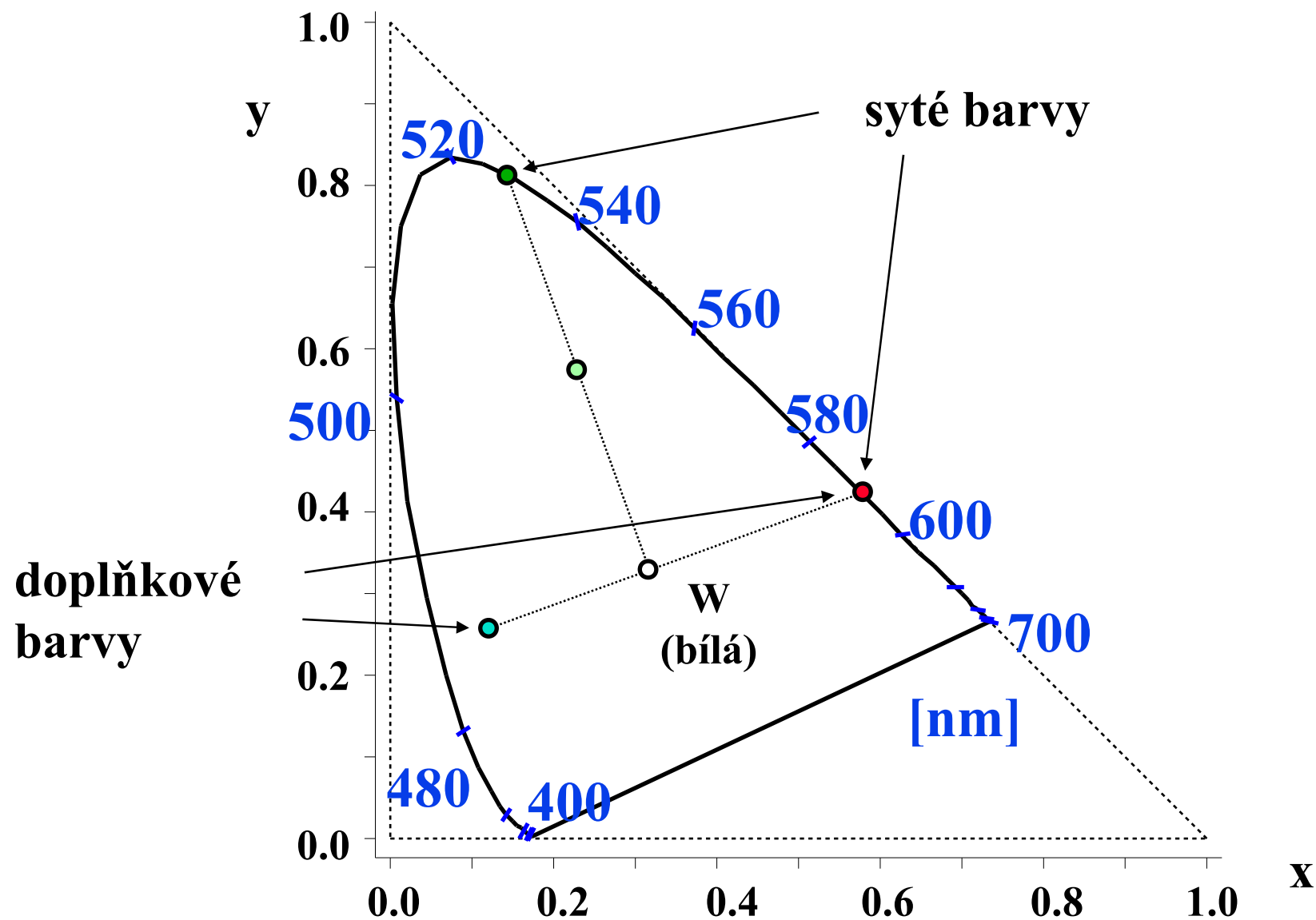


- základní **barevné složky**: červená, zelená, modrá
 - výhodné pro aktivní zobrazování (CRT monitor)
 - lidský zrakový systém vnímá podobně
- ◆ aditivní skládání barev
 - černé pozadí (nulová barva, vypnutý displej)
 - např. bílou dostaneme složením maxim všech tří složek

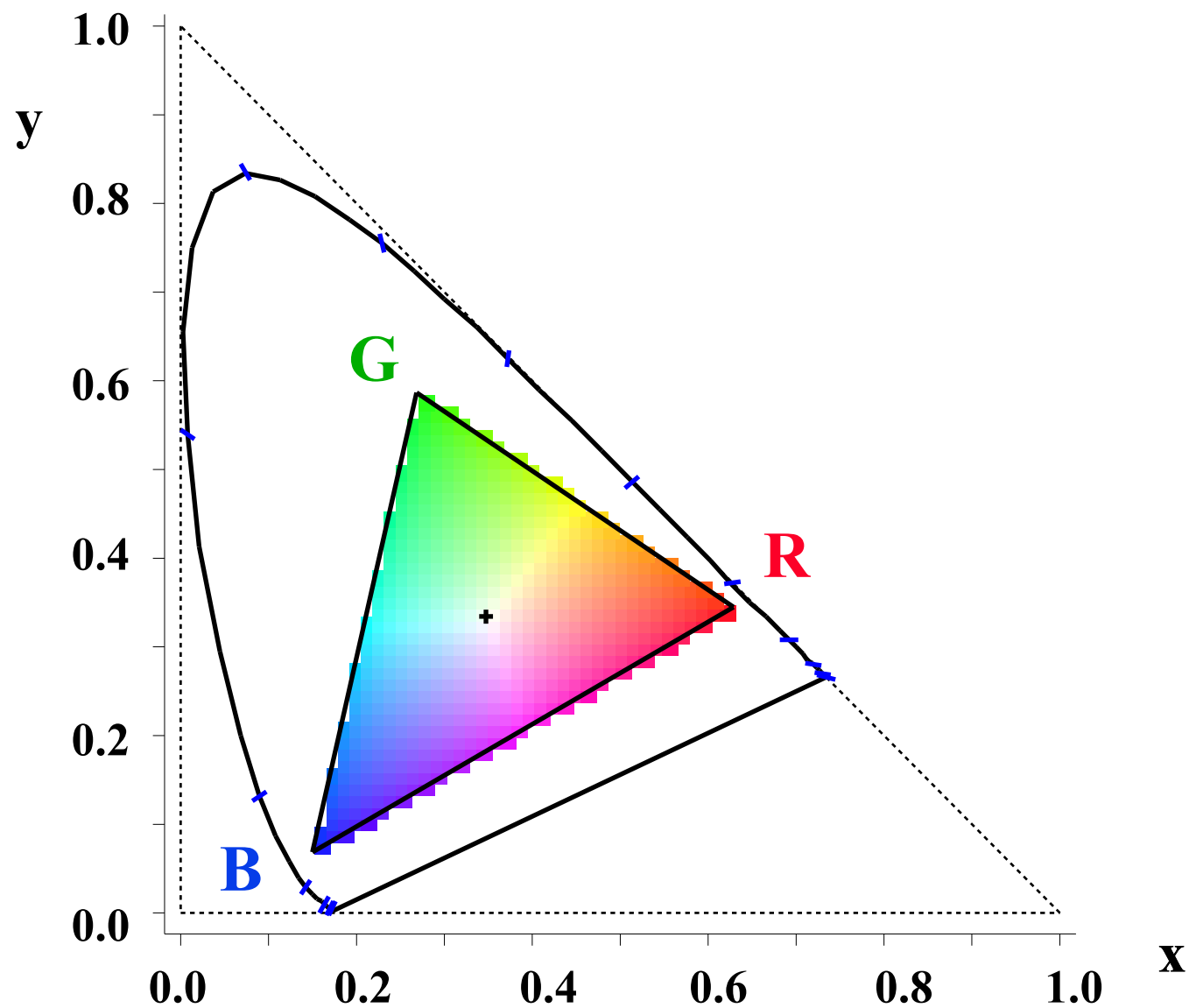
Aditivní skládání barev



Barevný prostor CIE-xy



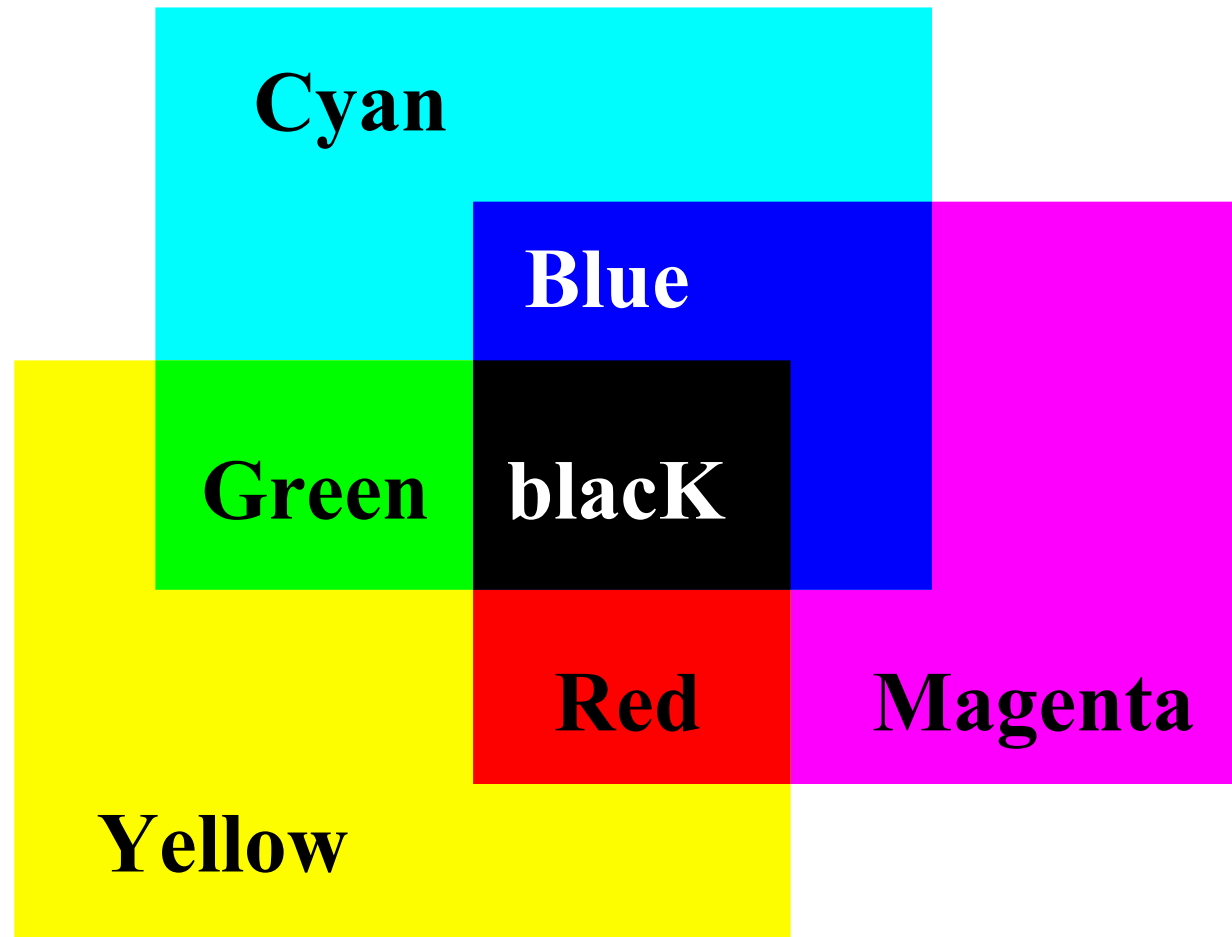
Gamut monitoru v CIE-xy



Barevný systém CMY(K)

- používá se při **tisku** a ve fotografii
 - tam, kde barevný dojem vzniká **pohlcením** některých složek bílého světla
- barvy se skládají **subtraktivně**
- základní barevná primitiva **C** (cyan), **M** (magenta), **Y** (yellow) odpovídají tiskařským barvám
 - **C, M, Y** jsou doplňkové k **R, G, B**

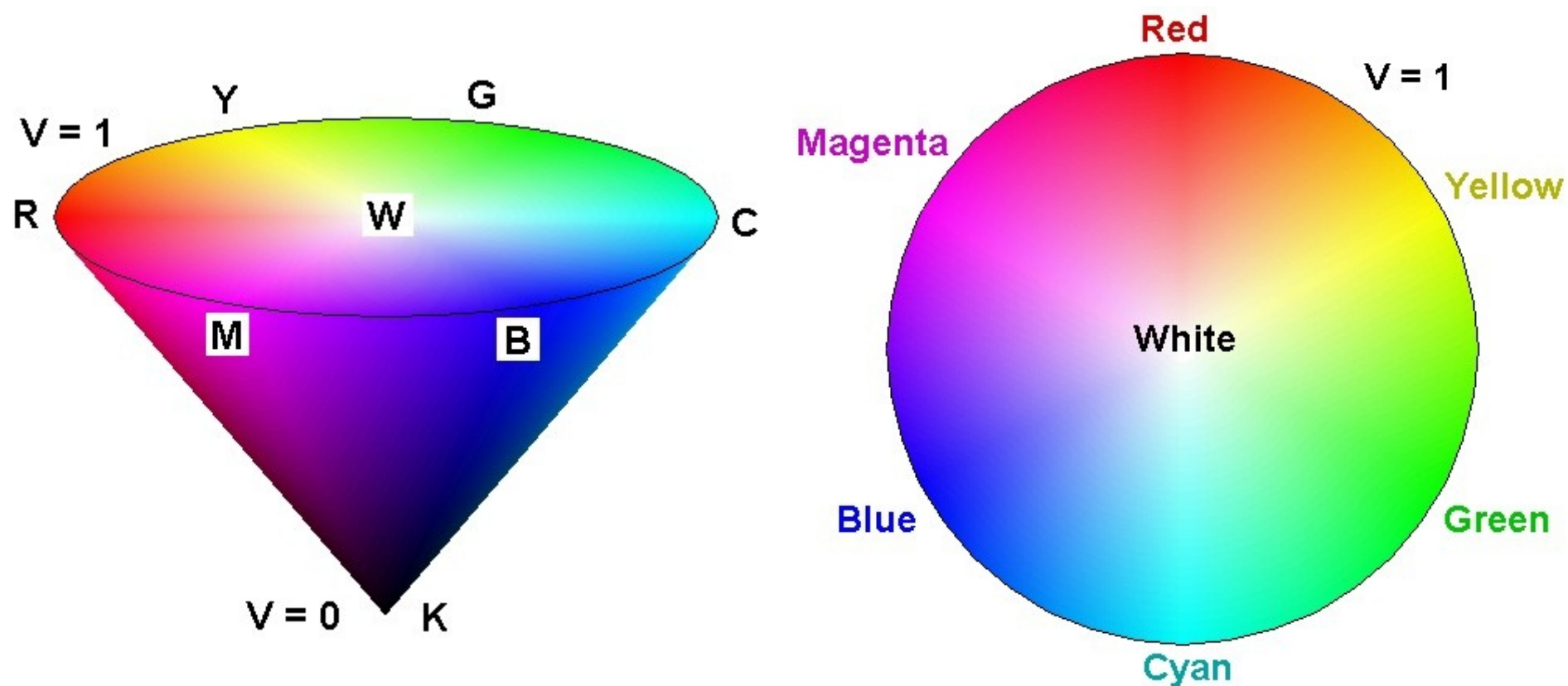
Subtraktivní skládání barev (CMY)



Barevný systém HSV

- orientovaný na **uživatele**
 - intuitivní veličiny: **barevný odstín** („hue”), **sytost** („saturation”) a **jas** („value”)
- význam jednotlivých složek:
 - **H**: základní spektrální barva (dominantní vlnová délka) – rozsah **0°** až **360°**
 - **S**: sytost, čistota barvy (poměr čisté barvy a bílé) – rozsah **0** (bílá) až **1** (spektrální barva)
 - **V**: jas, intenzita – rozsah **0** (černá) až **1**

Barevný kruh



Převod RGB → HSV

```
procedure RGB2HSV ( R,G,B : real; var H,S,V : real );  
var min, max, delta : real;  
begin  
  min := minimum(R,G,B); max := maximum(R,G,B);  
  V := max; delta := max - min;  
  if max <> 0.0 then S := delta/max  
    else S := 0.0;  
  if delta <> 0.0 then  
    begin                                     { chromatický případ }  
      if R = max then H := (G - B)/delta else  
      if G = max then H := 2 + (B - R)/delta  
      else H := 4 + (R - G)/delta;  
      H := H * 60.0;                          { převod na stupně }  
      if H < 0.0 then H := H + 360.0;  
    end;  
  end;
```

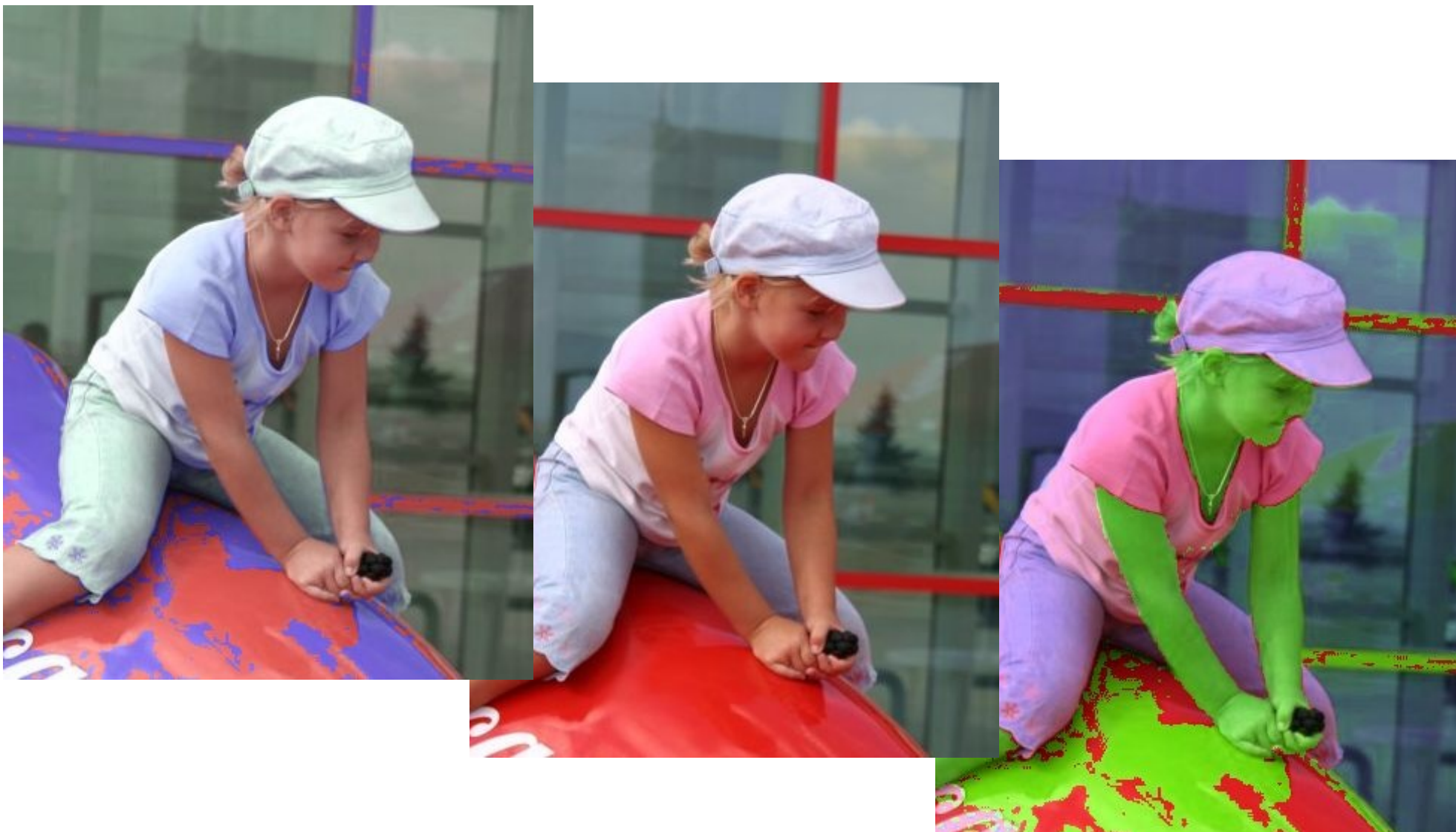
Barevné operace

- ♦ převod **RGB** $\rightarrow$ **HSV**
- ♦ manipulace se **sytností S**
- ♦ manipulace s **odstínem H**
 - ♦ změna barvy objektu
 - ♦ selektivní přebarvování ...
- ♦ převod zpět **H'S'V'** $\rightarrow$ **R'G'B'**

HSV operace



HSV operace



Příklady barevných operací



Příklady barevných operací



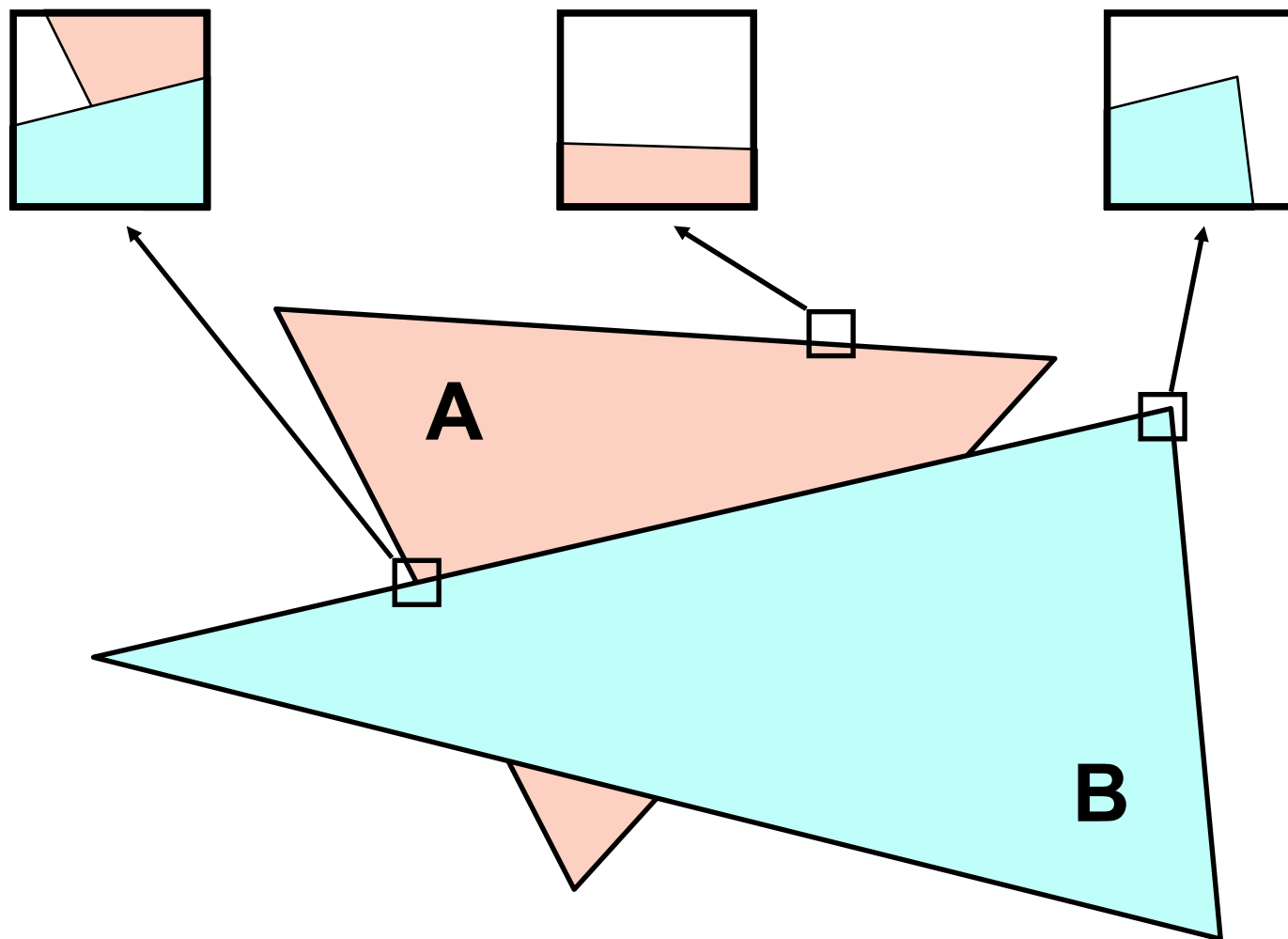
Průhlednost obrazu

- ❖ základy vnímání barev
- ❖ rastrový obraz a jeho pořizování
- ❖ barevné systémy, barevné operace
- ❖ **průhlednost a kompozice obrazu**
- ❖ histogramy a jasové operace
- ❖ konvoluce a lineární filtry, rank-filtry
- ❖ další efekty na rastrových obrázcích
- ❖ HDR grafika, pořizování a prohlížení HDRI

Kompozice obrázků

-  **montáž** několika reálných obrázků
 - vkládání objektů do jiného pozadí, ..
-  **prolínání** obrázků, „**fade-in**“, „**fade-out**“
 - animace, střih
-  **syntéza** obrazu
 - skládání umělého obrázku z několika samostatně vyrobených dílů (např.: pozadí, popředí, plameny, mlha, ..)

Pokrytí plochy pixelu



Kanál alfa

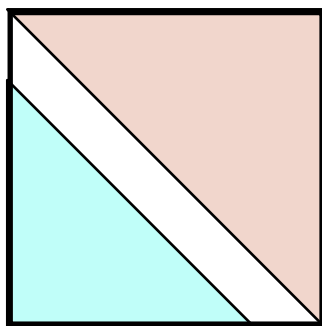
- ✦ procentuální **pokrytí pixelu** barvou
 - doplněk **průhlednosti**:
 - $\alpha = 0$... zcela průhledný pixel (nemá vliv na výsledek)
 - $\alpha = 1$... neprůhledný pixel („nic za ním neprosvítá”)
- ✦ ukládání hodnoty α v každém pixelu
 - často celočíselná reprezentace ($0 \div 255$)
 - čtveřice $[R, G, B, \alpha]$
 - často se ukládá již **přednásobená čtveřice** $[R\alpha, G\alpha, B\alpha, \alpha]$

Skládání dvou obrázků

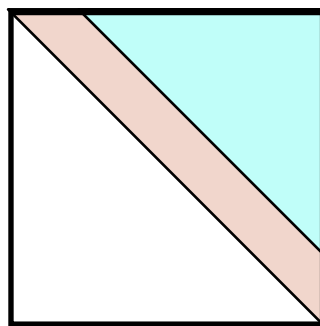
- ✦ dva skládané pixely $[A, \alpha_A]$ resp. $[B, \alpha_B]$
 - potřebuji určit výslednou hodnotu $[C, \alpha_C]$
- ✦ ? model pro skládání pixelů ?

$$\alpha_A = 0.5$$

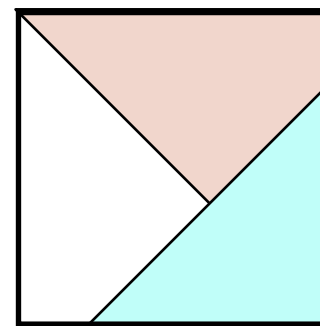
$$\alpha_B = 0.4$$



?



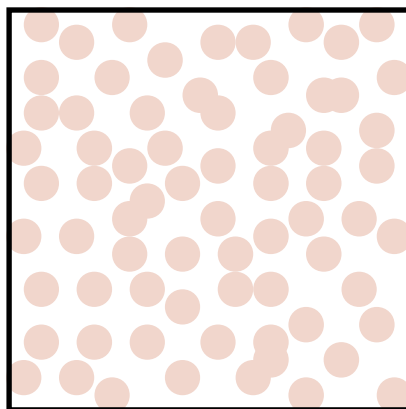
?



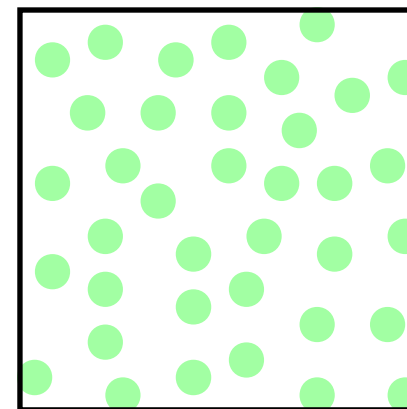
Model pokrytí pixelu

- pixel $[A, \alpha_A]$ je náhodně pokryt barvou A s rovnoměrně rozloženou pravděpodobností α_A
- skládání geometricky nezávislých tvarů
 - vyhovuje ve většině případů

$\alpha = 0.5$

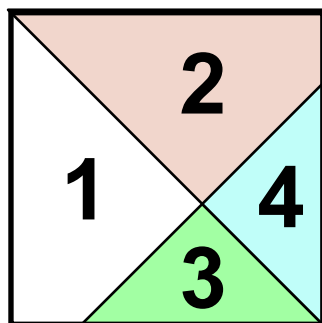


$\alpha = 0.2$



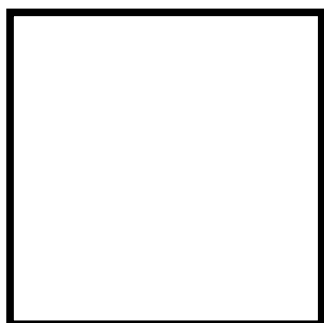
Překrytí dvou pixelů

	oblast	plocha	vybarvení
1	nic	$(1 - \alpha_A)(1 - \alpha_B)$	0
2	A	$\alpha_A(1 - \alpha_B)$	0, A
3	B	$\alpha_B(1 - \alpha_A)$	0, B
4	A i B	$\alpha_A\alpha_B$	0, A, B

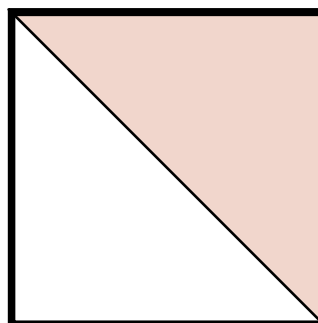


celkem 12 možností

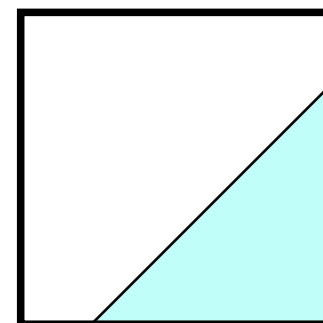
Binární operace I



nic
clear



A



B

barvy

$(0,0,0,0)$

$(0,A,0,A)$

$(0,0,B,B)$

F_A

0

1

0

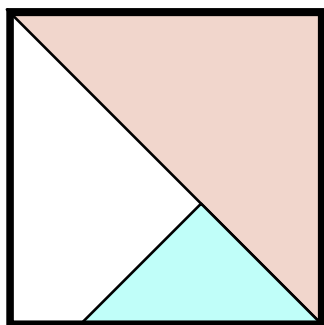
F_B

0

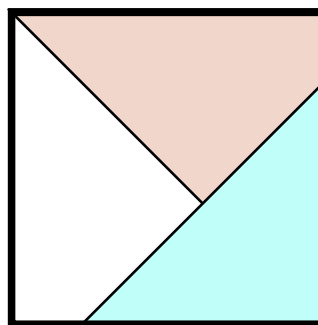
0

1

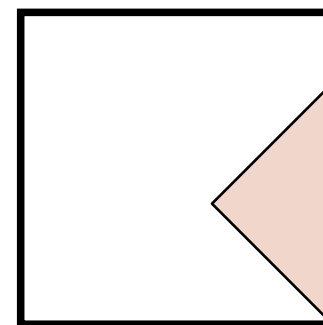
Binární operace II



A přes B
A over B



B přes A
B over A



A v B
A in B

barvy

(0,A,B,A)

(0,A,B,B)

(0,0,0,A)

F_A

1

$(1 - \alpha_B)$

α_B

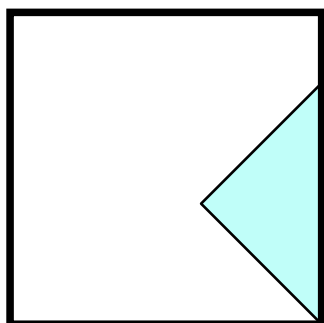
F_B

$(1 - \alpha_A)$

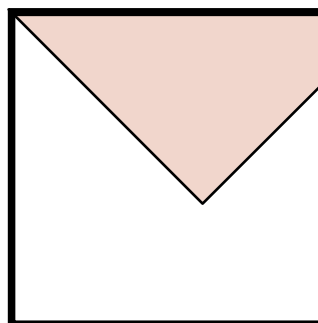
1

0

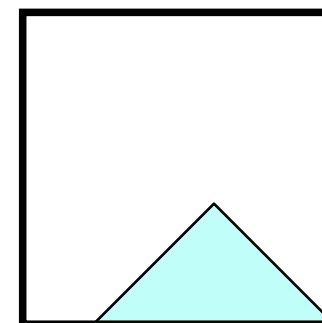
Binární operace III



B v A
B in A



A mimo B
A held out by B



B mimo A
B held out by A

barvy

(0,0,0,B)

(0,A,0,0)

(0,0,B,0)

F_A

0

$(1 - \alpha_B)$

0

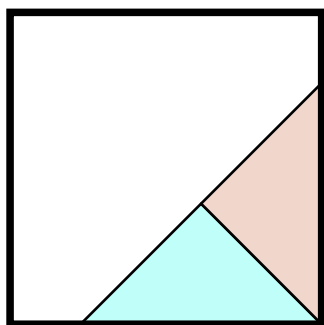
F_B

α_A

0

$(1 - \alpha_A)$

Binární operace IV



A na povrchu B
A atop B

barvy

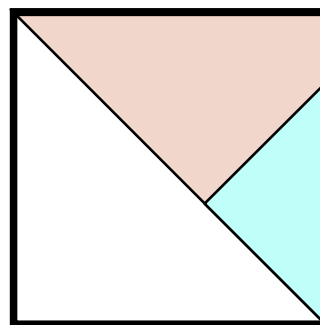
(0,0,B,A)

F_A

α_B

F_B

$(1 - \alpha_A)$

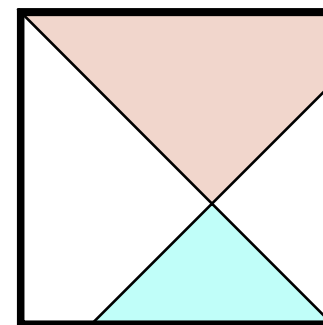


B na povrchu A
B atop A

(0,A,0,B)

$(1 - \alpha_B)$

α_A



A xor B

(0,A,B,0)

$(1 - \alpha_B)$

$(1 - \alpha_A)$

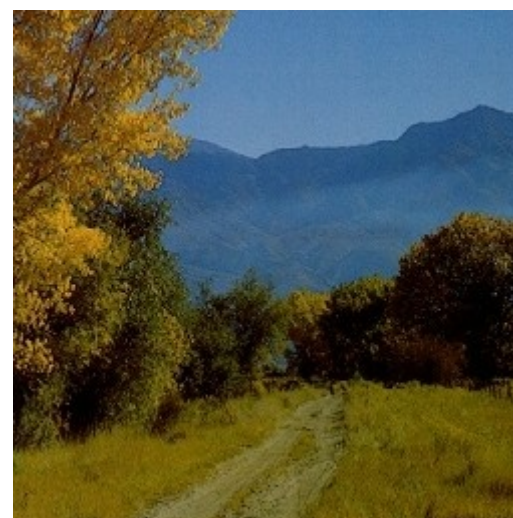
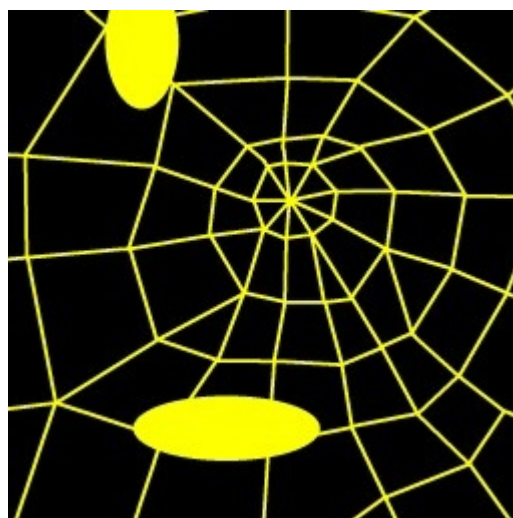
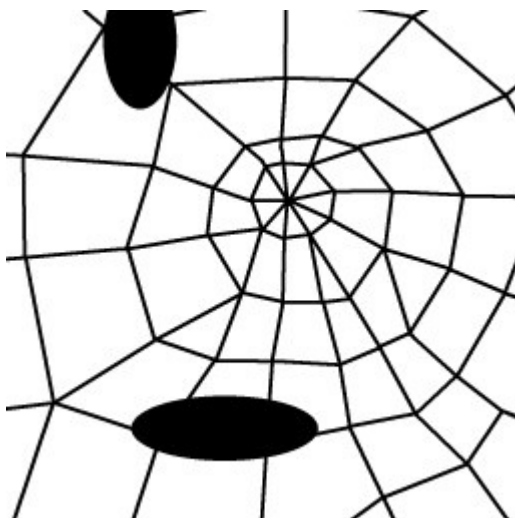
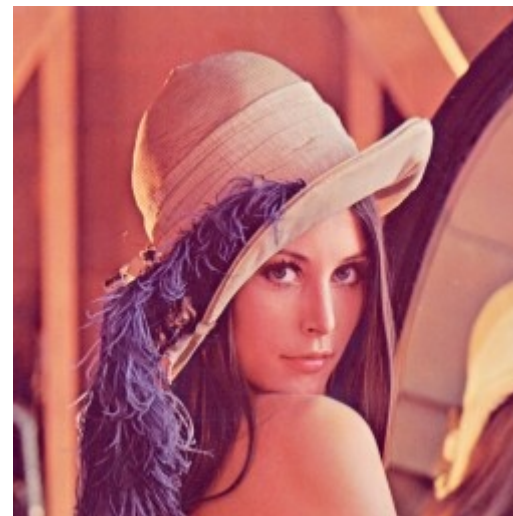
Aritmetika operací

-  operace skládání dvou pixelů A **op** B :
 $[F_A R_A + F_B R_B, F_A G_A + F_B G_B, F_A B_A + F_B B_B, F_A \alpha_A + F_B \alpha_B]$
-  operátor **darken** (A, ρ):
 $[\rho R_A, \rho G_A, \rho B_A, \alpha_A]$
-  operátor **fade** (A, δ):
 $[\delta R_A, \delta G_A, \delta B_A, \delta \alpha_A]$
-  operátor **opaque** (A, ω):
 $[R_A, G_A, B_A, \omega \alpha_A]$

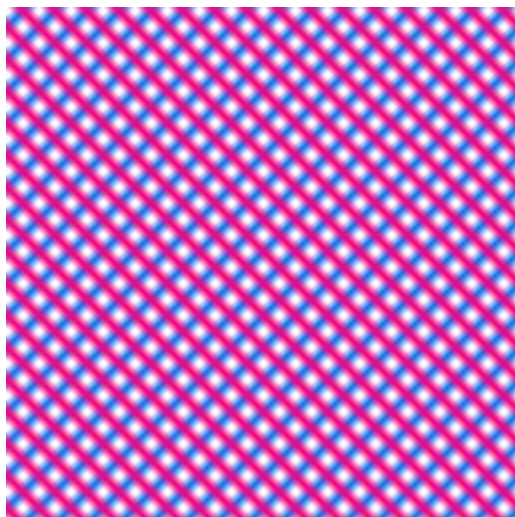
Plus, příklady

- operátor sčítání **A plus B**:
 $[R_A + R_B, G_A + G_B, B_A + B_B, \alpha_A + \alpha_B]$
- příklad 1: prolínání dvou obrázků
fade(A,t) plus fade(B,1 - t)
- příklad 2: hořící strom
**(FFire plus (BFire held out by Tree)) over darken(Tree,0.8)
over Background**

Ukázky – vstupy



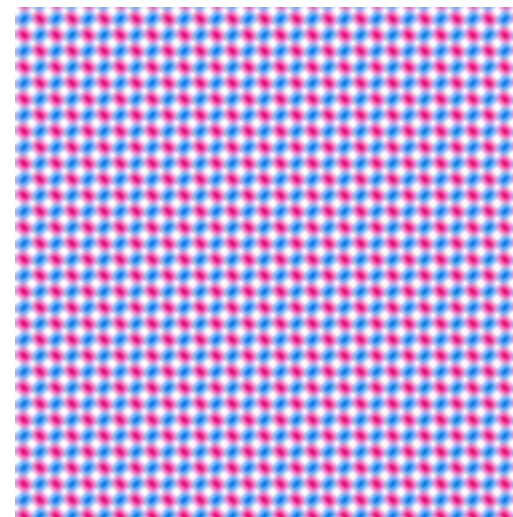
Ukázky – binární operace I



1 over 2



1 atop 2



1 xor 2

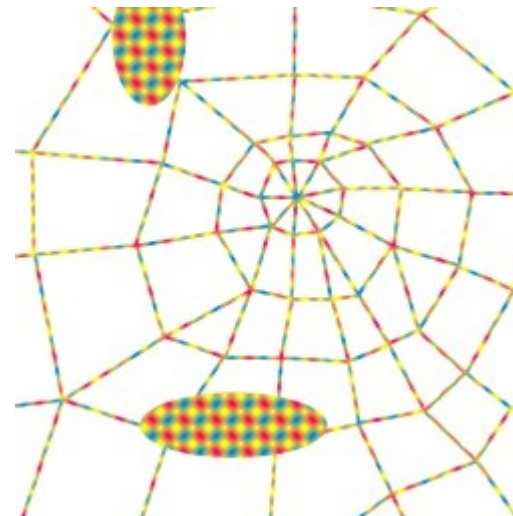
Ukázky – binární operace II



1 in 2

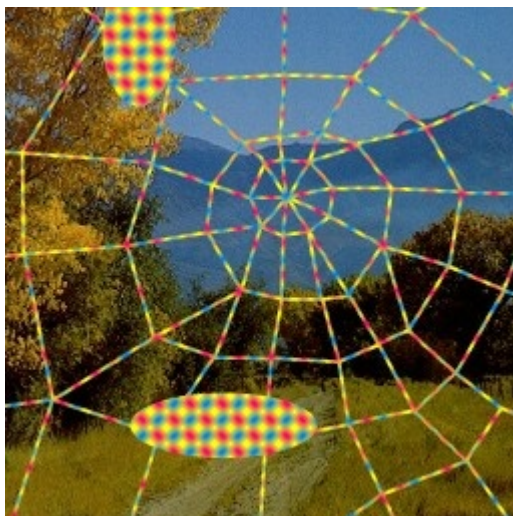


1 held out by 2



(1 xor 2) atop W

Ukázky – binární operace III



$((1 \text{ xor } 2) \text{ atop } W) \text{ over } V$

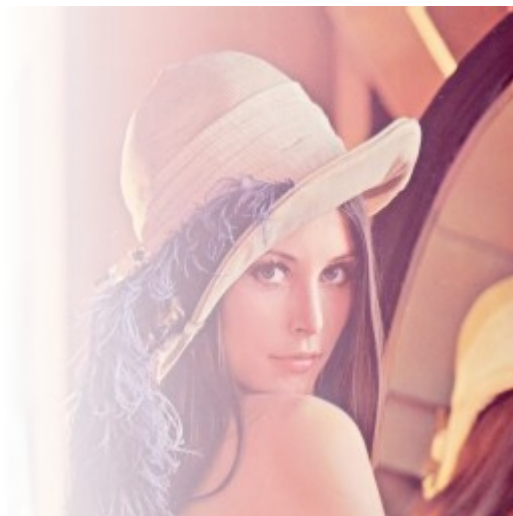


$(V \text{ atop } (1 \text{ xor } 2)) \text{ over } L$

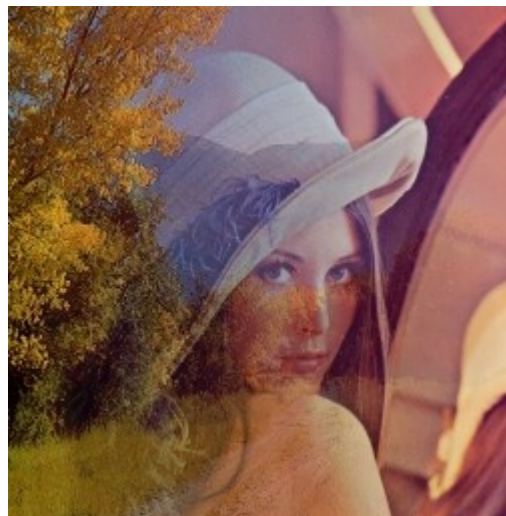


$V \text{ atop } (1 \text{ xor } 2)$

Ukázky – prolínání



fade(L, horiz)



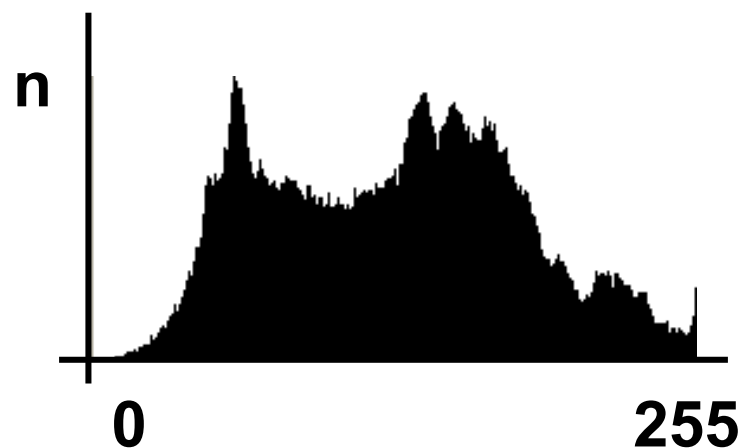
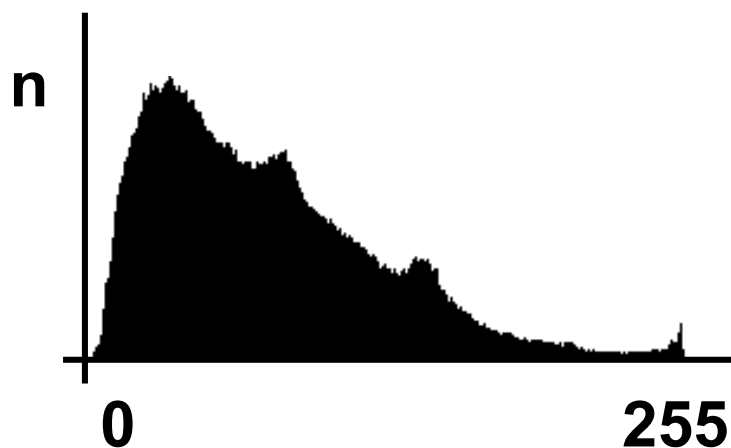
fade(L, horiz) over V

Histogramové operace

- ❖ základy vnímání barev
- ❖ rastrový obraz a jeho pořizování
- ❖ barevné systémy, barevné operace
- ❖ průhlednost a kompozice obrazu
- ❖ **histogramy a jasové operace**
- ❖ konvoluce a lineární filtry, rank-filtry
- ❖ další efekty na rastrových obrázcích
- ❖ HDR grafika, pořizování a prohlížení HDRI

Histogram obrázku

- ◆ **tabulka četností** jednotlivých jasových (barevných) hodnot
 - ◆ **spojitý případ** – hustota pravděpodobnosti
- ◆ přímé použití – fotografie

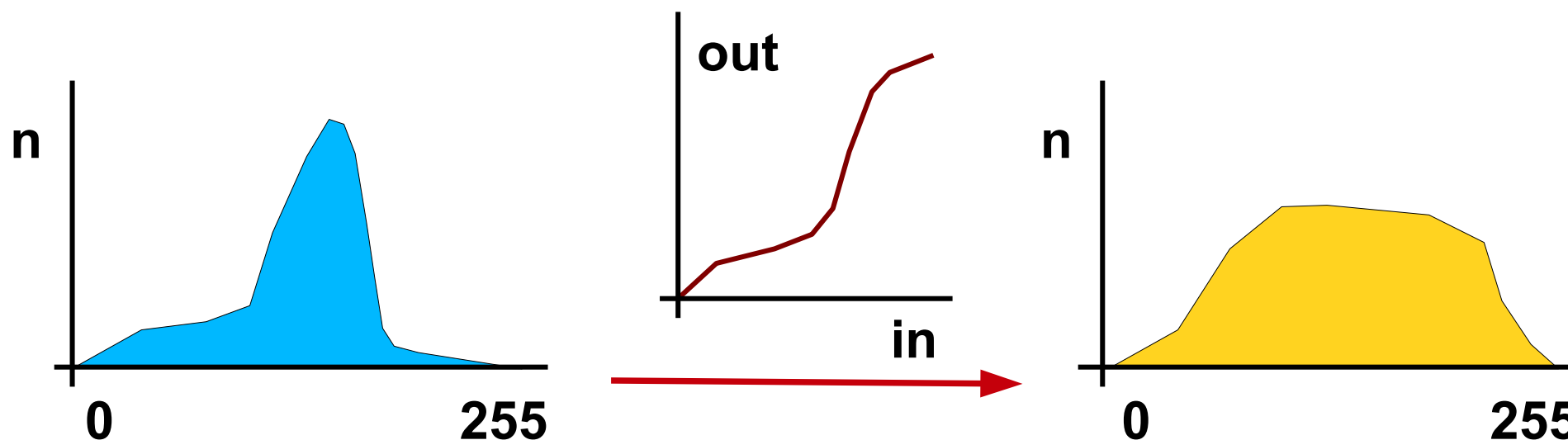


Jasové poměry obrázku

- ◆ histogram → první odhad **expozice** obrazu
- ◆ přeexponované nebo podexponované snímky
- ◆ nedostatečný nebo příliš velký **kontrast**
- ◆ „dobrý histogram“
 - ◆ obraz má odstíny ve všech částech škály
 - ◆ ~ detaily čitelné ve stínech i jasných partiích
- ➡ nedal by se špatný histogram „opravit“?

Jasová transformace

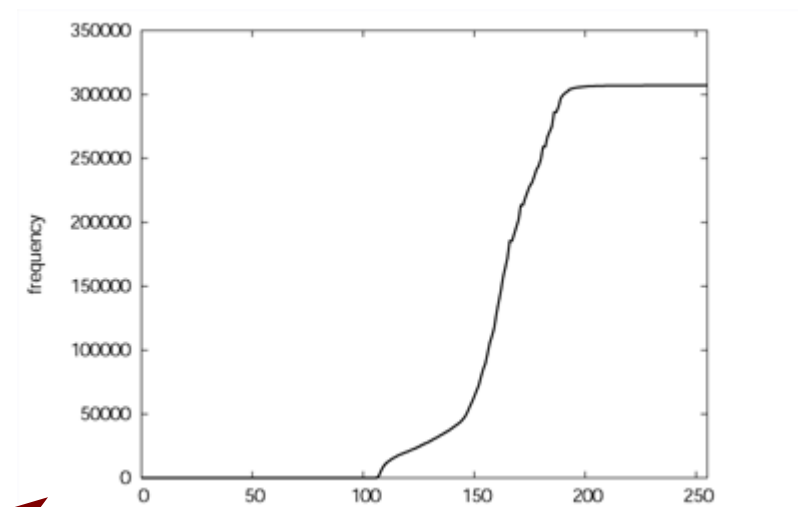
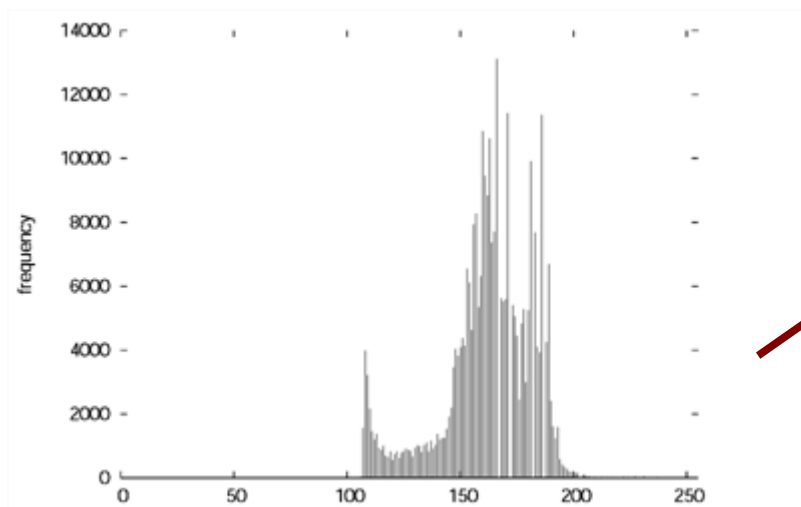
- ♦ **převodní funkce** („transfer function“) mezi jasy na vstupu a výstupu
 - ♦ $t: R \rightarrow R$ (obvyčejně $[0; 1] \rightarrow [0; 1]$)
- ♦ gamma-korekce
- ♦ zvětšování kontrastu



Ekvalizace histogramu

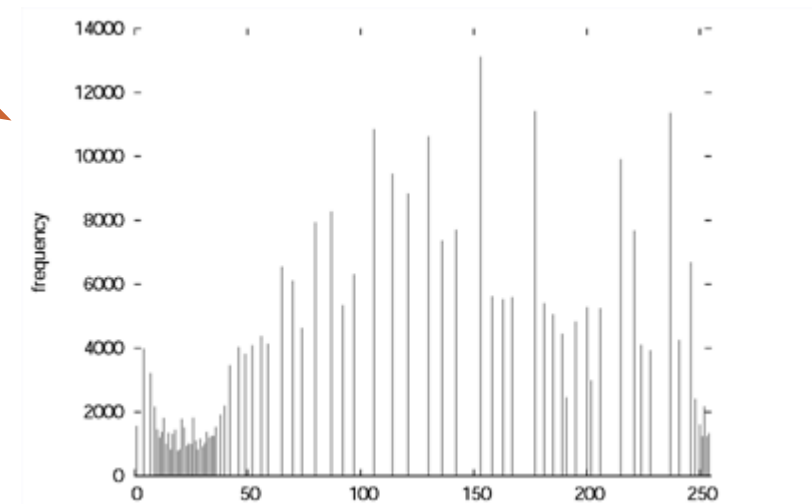
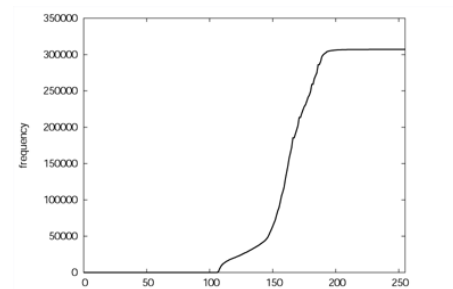
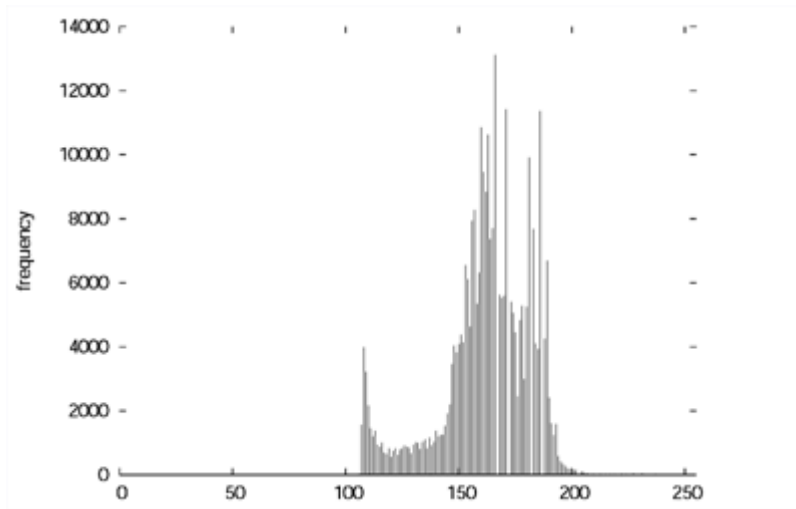
- ♦ umělá jasová transformace
 - ♦ snaží se o vyrovnaný histogram
- ♦ manipuluje s celými jasovými „sloupečky“
 - ♦ příp. rozděljuje četnější odstíny stochasticky
- ♦ **lokální ekvalizace** histogramu
 - ♦ analýzu dělá jen na (větším) okolí daného pixelu
 - ♦ může zlepšit čitelnost na celé ploše obrázku
 - ♦ nezachovává jednobarevné plochy!

Příklad globální ekvalizace

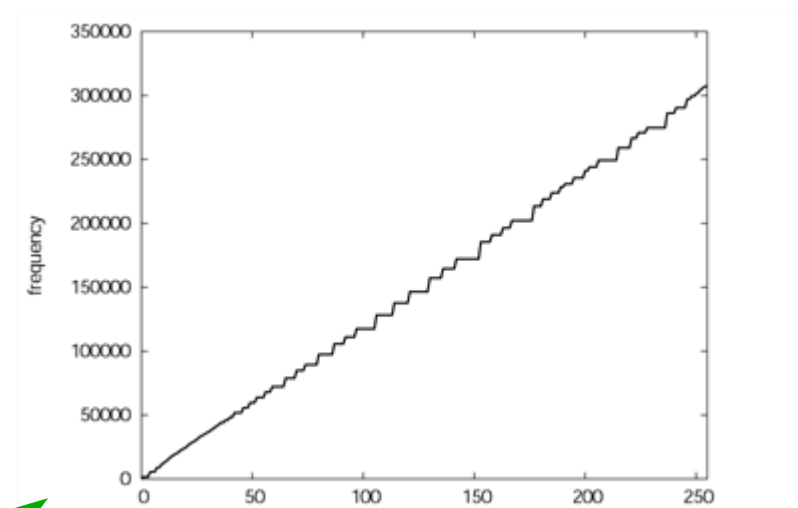
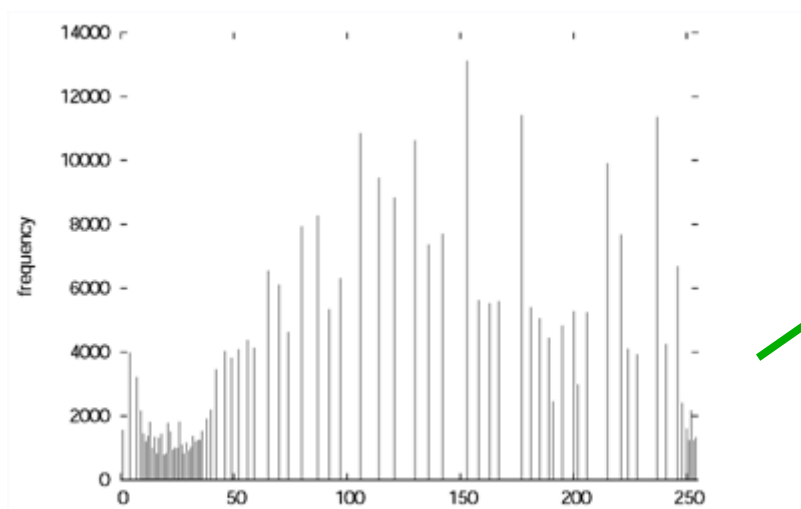


**kumulovaný histogram
(jasová transformace)**

Jasová transformace



Výsledek po ekvalizaci



kumulovaný histogram

Lineární filtry

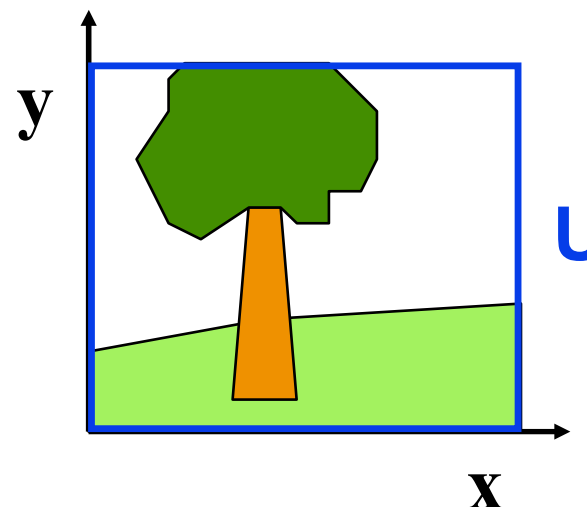
- ❖ základy vnímání barev
- ❖ rastrový obraz a jeho pořizování
- ❖ barevné systémy, barevné operace
- ❖ průhlednost a kompozice obrazu
- ❖ histogramy a jasové operace
- ❖ **konvoluce a lineární filtry, rank-filtry**
- ❖ další efekty na rastrových obrázcích
- ❖ HDR grafika, pořizování a prohlížení HDRI

Matematická definice obrazu

„obrazová funkce“

$$f: U \subset \mathbb{R}^2 \rightarrow \mathbb{R}^n$$

$$f: [x, y] \rightarrow [a_1, a_2, \dots, a_n]$$



poloha bodu
v rovině

atributy obrazu
(barva, průhlednost)

Konvoluce

- ♦ spojitá varianta „váženého klouzavého průměru“
 - ♦ váhová funkce **g**
- ♦ úzká souvislost s **Fourierovou transformací**
 - ♦ spektrální prostor
 - ♦ filtry typu „dolní propust“ apod.

$$(f * g)(x) = \int_{-\infty}^{\infty} f(t) \cdot g(x - t) dt$$

1D varianta

Diskrétní konvoluce

- ♦ „vážený klouzavý průměr“ na posloupnosti (tabulce)
 - ♦ váhová posloupnost (tabulka) **g**
- ♦ souvislost s **diskrétní Fourierovou transformací**

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m] \cdot g[n-m]$$

1D varianta

Účinky konvoluce

- ◆ dolní propust (jen kladné hodnoty **g**)
 - ◆ rozmazání obrazu
 - ◆ potlačení šumu
- ◆ horní propust (kladné i záporné hodnoty, součet **0**)
 - ◆ detekce hran v obraze
 - ◆ po modifikaci: ostření obrazu
- ◆ složitější spektrální filtry
- ◆ další efekty („emboss“, ...)

Rozmazání obrazu



originál



Gauss

1	2	1
2	4	2
1	2	1

/ 16

Detekce hran („high-pass filter“)



originál

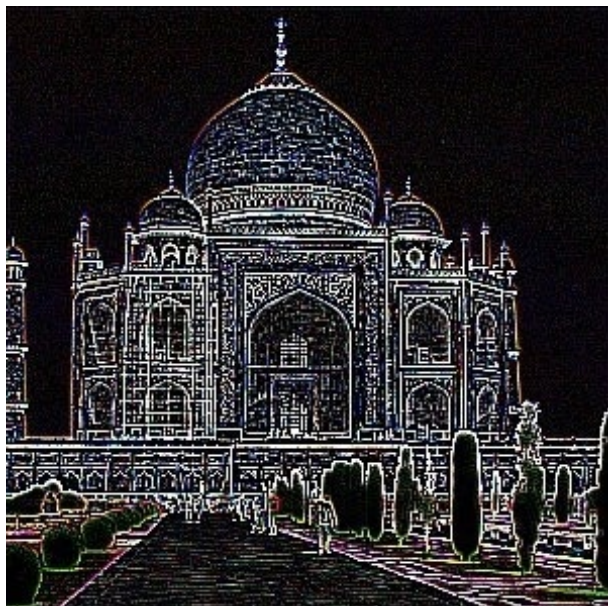


Sobel (2 směry)

1	2	1
0	0	0
-1	-2	-1

1	0	-1
2	0	-2
1	0	-1

Ostření obrazu



Laplacián

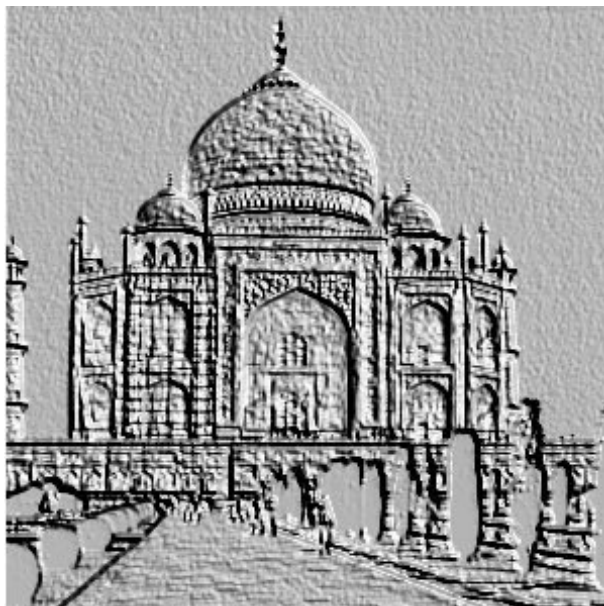
0	-1	0
-1	4	-1
0	-1	0



zaostření

0	-1	0
-1	5	-1
0	-1	0

„Emboss“ efekt



emboss



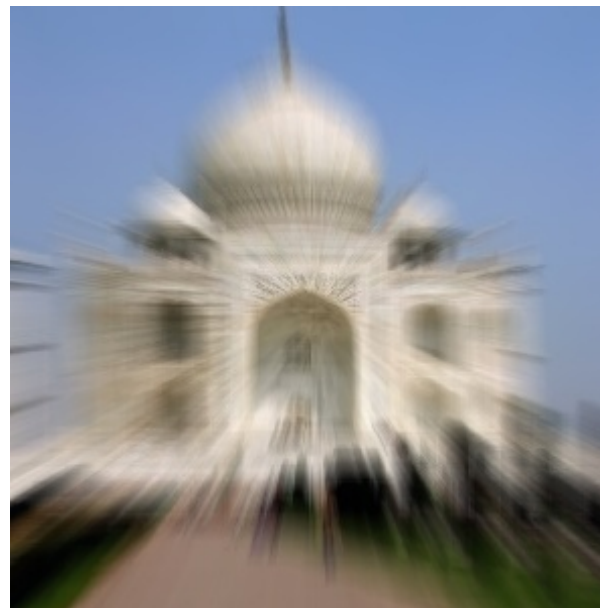
originál

-1	0	0
0	1	0
0	0	0

Neuniformní rozmazání



originál



radiální rozmazání
(1D rozmazání)

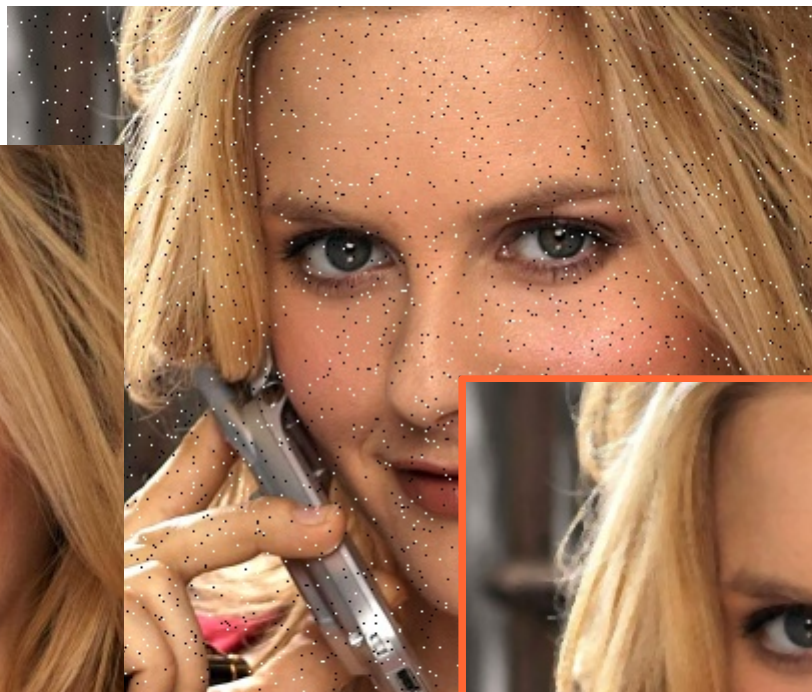
Nelineární filtry („rank filtry“)

- ♦ **okénkovací filtrace** (podobně jako u konvoluce)
- ♦ v okénku se hodnoty pixelů seřadí podle velikosti:
 - **medián** – potlačení šumu, umělecké efekty, ...
 - **minimum** – „eroze“
 - **maximum** – „dilatace“
- ♦ různé **tvary okénka**
 - ♦ čtverec
 - ♦ kruh
 - ♦ křížek (zachová ostré rohy)

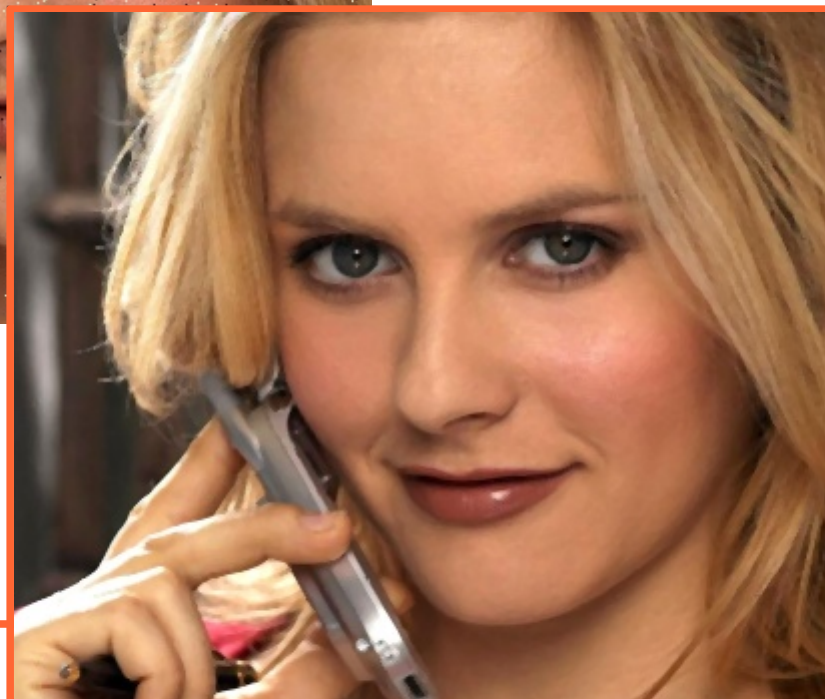
Medián na potlačení šumu



originál



pepř & sůl



medián 3×3

Dilatace a eroze



dilatace



eroze

Další efekty

- ◆ základy vnímání barev
- ◆ rastrový obraz a jeho pořizování
- ◆ barevné systémy, barevné operace
- ◆ průhlednost a kompozice obrazu
- ◆ histogramy a jasové operace
- ◆ konvoluce a lineární filtry, rank-filtry
- ◆ **další efekty na rastrových obrázcích**
- ◆ HDR grafika, pořizování a prohlížení HDRI

Potlačování šumu

- ◆ pokročilejší metody se snaží o **zachování hran** obrazu
 - ◆ nelze použít obyčejnou redukci vysokých frekvencí
- ◆ varianty **mediánového filtru**
- ◆ **neizotropické filtrování**
 - ◆ rozmazávání se děje ve směru „vrstevnic“ obrazu (kolmo na gradient obrazové funkce)
- ◆ filtrace **rotující maskou**
 - ◆ několik uvažovaných okolí daného pixelu
 - ◆ průměruje („mediánuje“) se v okénku s minimálním rozptylem

Umělecké efekty

- ♦ napodobení **malířských/kreslířských technik**
- ♦ **simulované tahy** štětcem/perem/pastelem
- ♦ efekty typu „**mozaika**“, vitráž, ...
- ♦ **NPR** (nefotorealistické) **efekty**
 - ♦ zvýrazňování hran
 - ♦ vyplňování vnitřních oblastí objektů
 - ♦ narůstání oblastí (aplikace segmentačních metod)
 - ♦ ...

Příklad – umělecký efekt

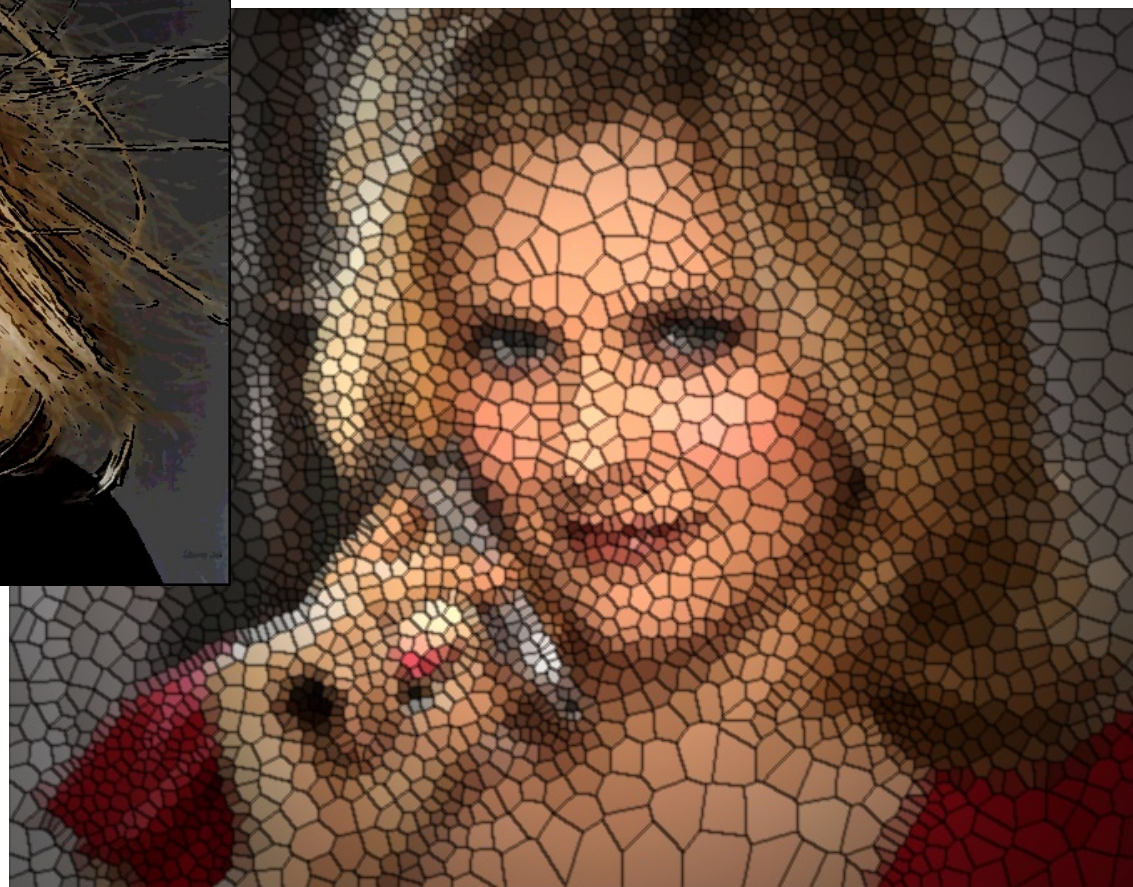


originál



kresba

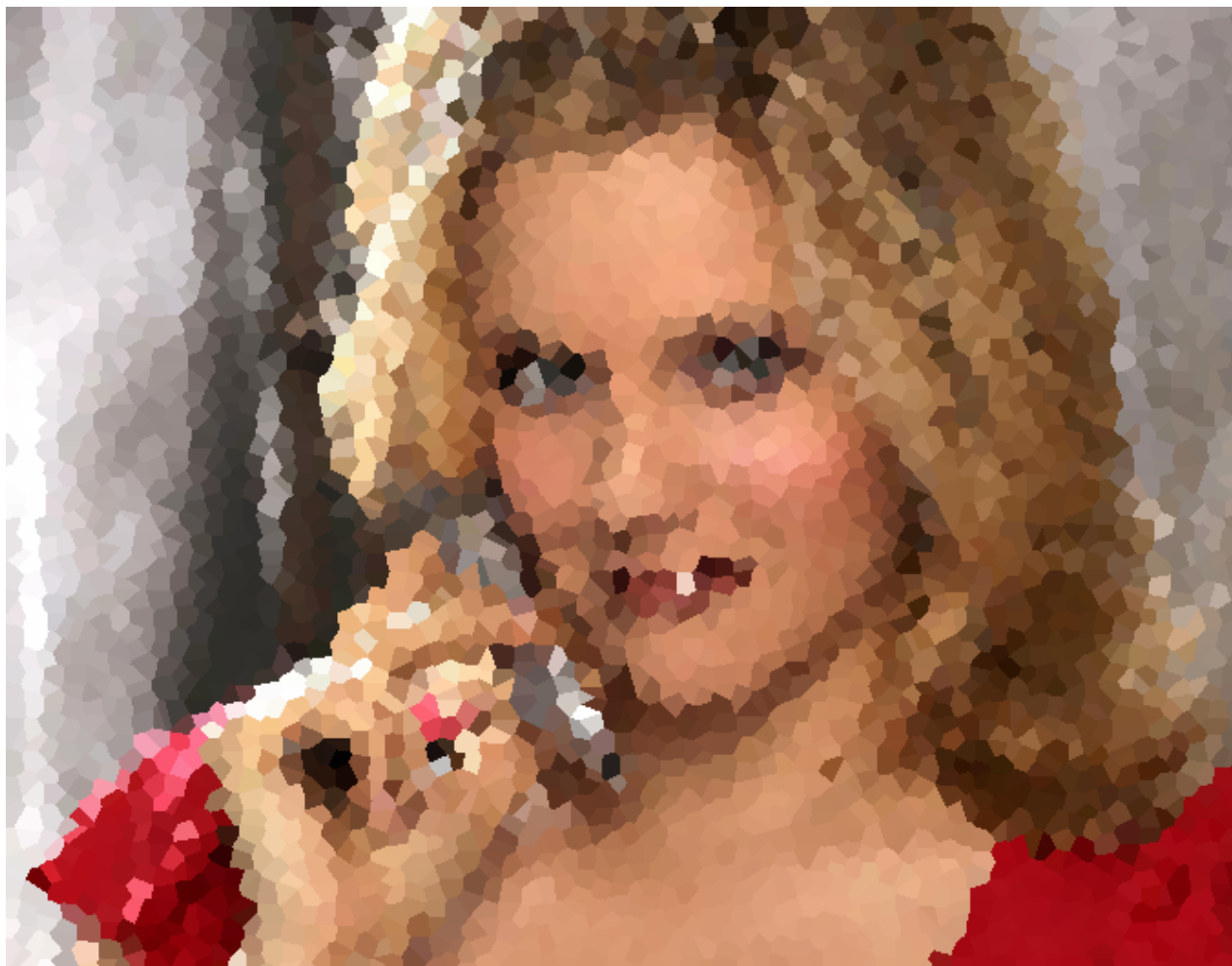
Příklad – NPR filtry



Příklad – NPR filtry



Příklad – mozaika



HDR grafika

- ❖ základy vnímání barev
- ❖ rastrový obraz a jeho pořizování
- ❖ barevné systémy, barevné operace
- ❖ průhlednost a kompozice obrazu
- ❖ histogramy a jasové operace
- ❖ konvoluce a lineární filtry, rank-filtry
- ❖ další efekty na rastrových obrázcích
- ❖ **HDR grafika, pořizování a prohlížení HDRI**

HDR grafika

- ◆ **High Dynamic Range**
 - velký dynamický rozsah
 - pixely – čísla v **plovoucí desetinné čárce**
 - např. **float[3]** pro RGB (96bpp)
- ◆ **pořizování HDR dat**
 - výpočet (rendering)
 - fotografování (vícenásobná expozice)
- ◆ **zobrazování na LDR zařízení**
 - převod do normální škály („tone-mapping“)

Formát pixelu RGBE (Radiance)

- ❖ formát souboru **.hdr (Radiance)**
 - ❖ úsporné uložení (jen 4 byty na pixel)
 - individuální **mantisa** [RGB], společný **exponent** [E]
- ❖ **mantisa [RGB]**
 - typ **float**, normalizován mezi $\frac{1}{2}$ a 1 (maximální složka)
- ❖ **exponent [E]**
 - binární exponent v doplňkovém kódu (8-bitové číslo)
- ❖ příklad: [0.3, 0.02, 0.1]
 - = [0.6, 0.04, 0.2] $\cdot 2^{-1} \rightarrow$ [153, 10, 51, 127]

Další HDR formáty

◆ OpenEXR (.exr)

- Industrial Light & Magic (G. Lucas 1975, Star Wars etc.)
- zcela otevřený, knihovny jsou open-source
- různé typy komprese (ZIP, wavelets), používá často **half**
- uživatelsky **rošiřitelný formát pixelu**

◆ Portable Float Map (PFM)

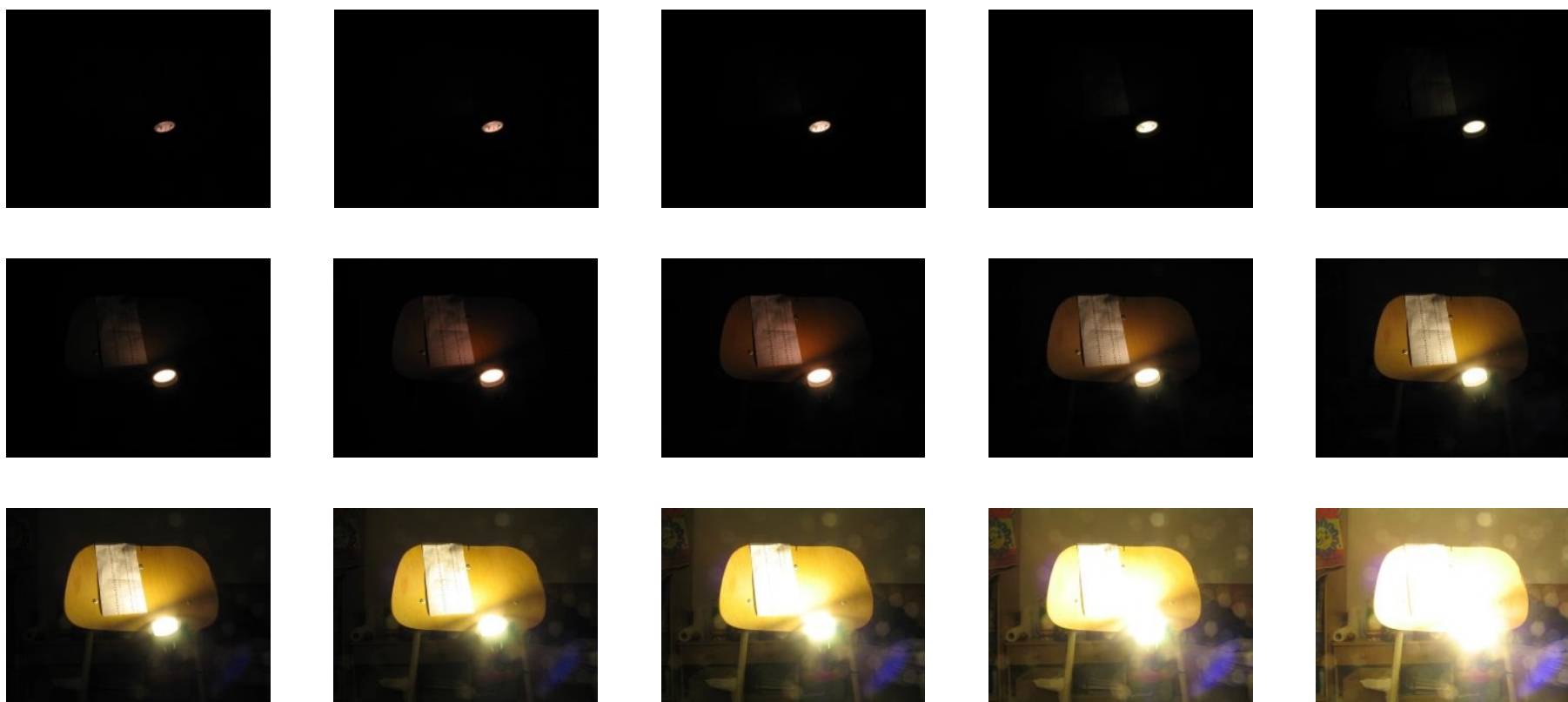
- analogický PPM / PGM / PBM
- pixely jsou tři čísla typu **float**
- bez komprese

Fotografování HDR

- ❖ **vícenásobná expozice**
 - statická scéna
 - konstantní clona, proměnlivý čas
 - sekvence např. od **1/1000s** do **2s**
 - vestavěný „bracketing“ (−2 EV, 0, +2 EV)
 - „super-bracketing“ (až 7 expozic rychle za sebou)
- ❖ **zpracování sekvence obrázků** do jednoho HDR
 - běžné programy na zpracování foto (PhotoShop, GiMP..)
 - HDR Shop (<http://www.hdrshop.com/>)
 - funkce: registrace snímků, auto-kalibrace

Příklad pořízení HDRI

- 15 snímků mezi 1/2000s a 8s (po 1 EV)
- sestavení ... HDR Shop



Reprodukce HDR

- ♦ jednoduché oříznutí dynamiky
 - přetečení → přezáření (bílá nebo nějaké „glare“ efekty)
- ♦ „tone mapping“
 - obecně: transformace **celé** originální škály do LDR
 - globální vs. lokální zobrazení
 - lokální zachování kontrastu, apod.



Aplikace HDR

◆ fotografie

- lepší reprodukce přirozené dynamiky scény
- citlivý „tone mapping“, výsledek nepůsobí nepřirozeně
- HDR panoramata (slunce v záběru, obloha vs. terén)

◆ CGI (počítačem generovaná grafika)

- dobře použitelná data pro „**environment mapping**“ (světelná mapa okolí)
- všechny interní výpočty a mezivýsledky jsou „HDR“
- realisticky vypadající lesklé odrazy, rozmazání pohybem, apod.

Příklady – „tone-mapping“



LDR



Tone-mapped HDR

Příklady – „tone-mapping“



Příklady – „tone-mapping“



Méně citlivý „tone-mapping“



Příklad s odrazem světla

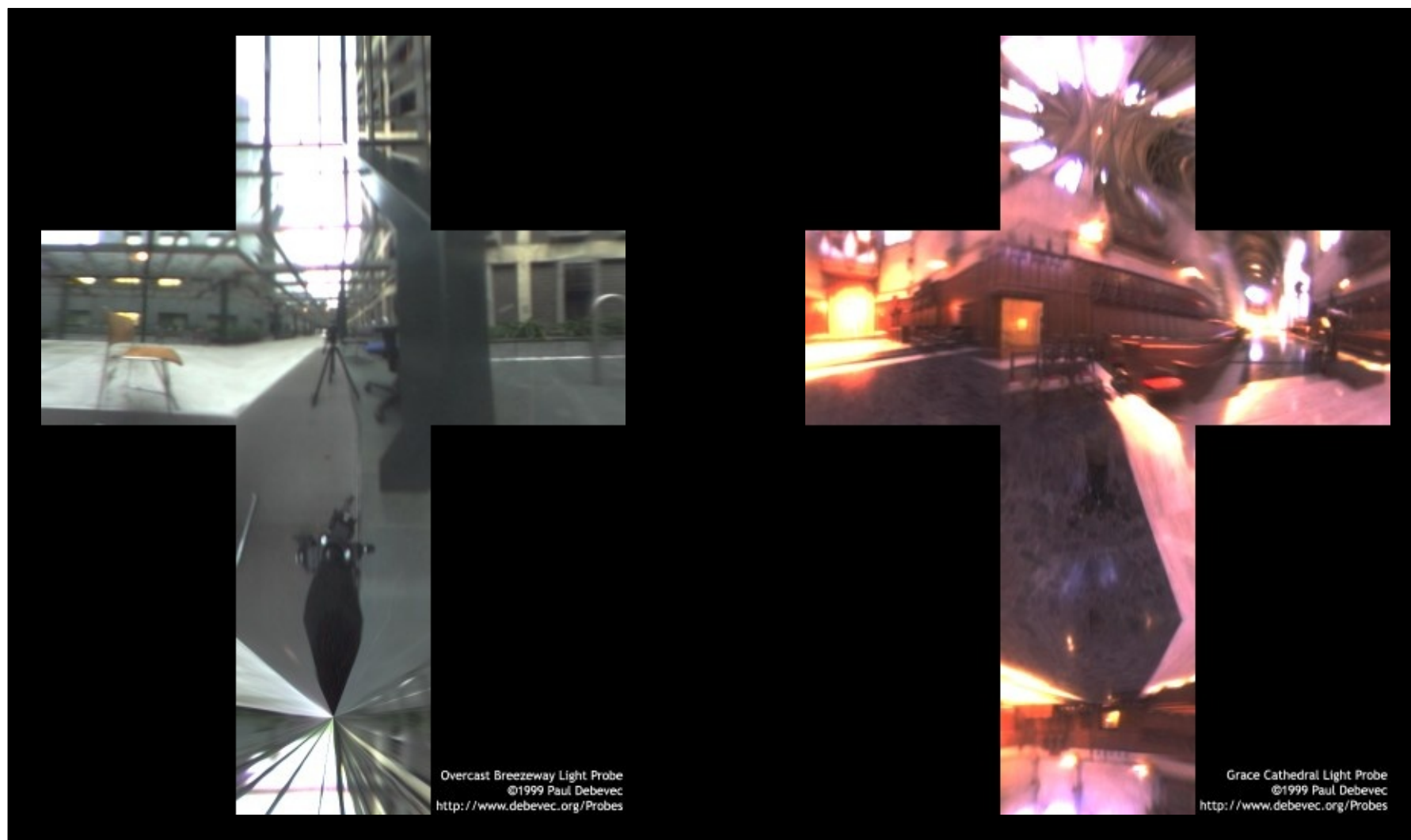


Environment-map (latitude/longitude mapping)

Příklad s odrazem světla



Environment („cube-map“)



Literatura

- ♦ Pratt W. K.: Digital Image Processing: PIKS Inside, 3rd Edition, Wiley-Interscience, 2001
- ♦ Gonzales R. C, Woods R. E.: Digital Image Processing, 3rd Edition, Prentice Hall, 2007
- <http://www.debevec.org/>
- <http://cgg.mff.cuni.cz/~pepca/hdr/>
- <http://www.hdrshop.com/>
- <http://www.mpi-inf.mpg.de/resources/hdr/>

Úlohy k přednášce ze zpracování obrazu (3. 11. 2010)

Budeme pracovat s **rastrovým obrázkem** obsahujícím pouze **dvě různé hodnoty jasu: 0 a 1**. Pozadí je černé (0) a objekty mají bílou barvu (1). Standardně obsahuje obrázek mnoho různých **obdélníků** s vodorovnými a svislými hranami (tj. rovnoběžnými s osami x a y). Obdélníky jsou nejenom disjunktní, ale můžete předpokládat, že jsou od sebe dostatečně daleko, tak, jak to konkrétní otázka bude potřebovat (tj. při řešení se nikdy nemusíte zabývat interakcí s více než jedním obdélníkem..). Dále se vůbec nemusíte zabývat okraji obrázku. Předpokládejte, že okraje jsou dostatečně oddělené od všech zajímavých útvarů (obdélníků, šumu atp.).

Úloha 1:

Obrázek je znečištěn **šumem typu „pepř a sůl“** - náhodně jsou některé samostatné pixely chybové, tedy obsahují nesprávnou hodnotu (bílou místo černé nebo naopak). Předpokládáme tak nízkou pravděpodobnost této chyby, že se nikdy dvě chyby nevyskytnou moc blízko sebe. Pro potlačení šumu budeme používat **mediánový okénkovací filtr**. Navrhněte **nejmenší velikost a tvar okénka**, aby filtr odstranil všechny chyby a přitom vůbec neporušil původní data! Pozor zejména na rohy obdélníků!

Úlohy k přednášce ze zpracování obrazu (3. 11. 2010)

Budeme pracovat s **rastrovým obrázkem** obsahujícím pouze **dvě různé hodnoty jasu: 0 a 1**. Pozadí je černé (0) a objekty mají bílou barvu (1). Standardně obsahuje obrázek mnoho různých **obdélníků** s vodorovnými a svislými hranami (tj. rovnoběžnými s osami x a y). Obdélníky jsou nejenom disjunktní, ale můžete předpokládat, že jsou od sebe dostatečně daleko, tak, jak to konkrétní otázka bude potřebovat (tj. při řešení se nikdy nemusíte zabývat interakcí s více než jedním obdélníkem..). Dále se vůbec nemusíte zabývat okraji obrázku. Předpokládejte, že okraje jsou dostatečně oddělené od všech zajímavých útvarů (obdélníků, šumu atp.).

Úloha 2:

Měli bychom trochu nepříjemnější „šum“ - chybové flíčky mohou být až **2x2 pixely** velké. Opět **navrhněte co nejmenší okno** pro mediánový filtr, který se má takového šumu zbavit. Chceme původní objekty co nejméně porušit, pokud by to šlo, tak úplně zachovat..

Úlohy k přednášce ze zpracování obrazu (3. 11. 2010)

Budeme pracovat s **rastrovým obrázkem** obsahujícím pouze **dvě různé hodnoty jasu: 0 a 1**. Pozadí je černé (0) a objekty mají bílou barvu (1). Standardně obsahuje obrázek mnoho různých **obdélníků** s vodorovnými a svislými hranami (tj. rovnoběžnými s osami x a y). Obdélníky jsou nejenom disjunktní, ale můžete předpokládat, že jsou od sebe dostatečně daleko, tak, jak to konkrétní otázka bude potřebovat (tj. při řešení se nikdy nemusíte zabývat interakcí s více než jedním obdélníkem..). Dále se vůbec nemusíte zabývat okraji obrázku. Předpokládejte, že okraje jsou dostatečně oddělené od všech zajímavých útvarů (obdélníků, šumu atp.).

Úloha 3:

Potřebujeme v obrázku **detekovat** pouze **vodorovné hrany objektů** (obdélníků). **Navrhňte nejjednodušší filtr včetně tvaru okénka**, který by to dokázal. Detekcí se rozumí označení každého pixelu hrany – např. na vodorovné hraně (nebo někde těsně vedle) budou ve výsledku jedničky, všude jinde nuly. Předpokládejte ideální vstupní obrázek – zcela bez šumu.

Pozn.: tentokrát přemýšlejte o lineárním (konvolučním) filtru, příp. modifikovaným nějakou další funkcí, např. absolutní hodnotou apod.

Úlohy k přednášce ze zpracování obrazu (3. 11. 2010)

Budeme pracovat s **rastrovým obrázkem** obsahujícím pouze **dvě různé hodnoty jasu: 0 a 1**. Pozadí je černé (0) a objekty mají bílou barvu (1). Standardně obsahuje obrázek mnoho různých **obdélníků** s vodorovnými a svislými hranami (tj. rovnoběžnými s osami x a y). Obdélníky jsou nejenom disjunktní, ale můžete předpokládat, že jsou od sebe dostatečně daleko, tak, jak to konkrétní otázka bude potřebovat (tj. při řešení se nikdy nemusíte zabývat interakcí s více než jedním obdélníkem..). Dále se vůbec nemusíte zabývat okraji obrázku. Předpokládejte, že okraje jsou dostatečně oddělené od všech zajímavých útvarů (obdélníků, šumu atp.).

Úloha 4:

Potřebovali bychom **detekovat** pouze **rohy obdélníků**, na hranách nebo na souvislých plochách nechceme mít žádnou odezvu. *Navrhněte co nejmenší filtr spolu s tvarem okénka*. Opět předpokládejte ideální vstupní obrázek – zcela bez šumu. Stejně jako v předchozím případě nemusí být nutně „odezva na roh“ přesně v tom rohu, stačí někde poblíž, pokud vám toto zmírnění podmínek ulehčí práci.

Úlohy k přednášce ze zpracování obrazu (3. 11. 2010)

Budeme pracovat s **rastrovým obrázkem** obsahujícím pouze **dvě různé hodnoty jasu: 0 a 1**. Pozadí je černé (0) a objekty mají bílou barvu (1). Standardně obsahuje obrázek mnoho různých **obdélníků** s vodorovnými a svislými hranami (tj. rovnoběžnými s osami x a y). Obdélníky jsou nejenom disjunktní, ale můžete předpokládat, že jsou od sebe dostatečně daleko, tak, jak to konkrétní otázka bude potřebovat (tj. při řešení se nikdy nemusíte zabývat interakcí s více než jedním obdélníkem..). Dále se vůbec nemusíte zabývat okraji obrázku. Předpokládejte, že okraje jsou dostatečně oddělené od všech zajímavých útvarů (obdélníků, šumu atp.).

Úloha 5:

Tentokrát to nebude filtr, ale algoritmus: **navrhněte co nejrychlejší algoritmus** (postup), který spočítá, **kolik obdélníkových objektů** se na obrázku vyskytuje. Složitost se měří počtem přečtených (+modifikovaných) pixelů obrázku.

Pokud chcete mít úlohu ještě těžší, můžete zkusit spočítat asymptotickou složitost vašeho algoritmu, tj. nějaký odhad, kolik přibližně operací potřebujete pro vstupní obrázek velikosti $M \times N$.

Pozn.: algoritmus nemůže mít menší složitost než $O(MN)$, protože musí každý vstupní pixel přečíst minimálně jednou..

Pozn.2: abyste nedělali triky, tak přidáme podmínku, že pokud použijete nějaké pomocné pole, tak se přístup do něj počítá stejně jako přístup do vstupního obrázku (abyste na začátku obrázek nekopírovali k sobě a pak netvrdili, že máte složitost přesně $M \times N$).